



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ**

**ÚSTAV RADIOELEKTRONIKY**

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF RADIO ELECTRONICS

## **DETEKCE OBLIČEJE V OBRAZE**

FACE DETECTION IN IMAGE

**BAKALÁŘSKÁ PRÁCE**

BACHELOR'S THESIS

**AUTOR PRÁCE**

AUTHOR

**TOBIÁŠ MALACH**

**VEDOUcí PRÁCE**

SUPERVISOR

**Ing. MARTIN MÉZL**

BRNO 2011



**VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ**

**Fakulta elektrotechniky  
a komunikačních technologií**

**Ústav radioelektroniky**

# Bakalářská práce

bakalářský studijní obor  
**Elektronika a sdělovací technika**

**Student:** Tobiáš Malach

**ID:** 115222

**Ročník:** 3

**Akademický rok:** 2010/2011

**NÁZEV TÉMATU:**

## Detekce obličeje v obraze

### POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s metodami detekce obličeje v obraze a proveďte teoretickou rešerši jednotlivých metod. Vyberte vhodnou metodu pro detekci obličeje z hlediska implementace v programovém prostředí Matlab. Zvolenou metodu implementujte a otestujte na databázi vytvořených snímků. Součástí práce bude také uživatelské rozhraní (GUI) pro demonstraci funkčnosti uvedené metody. Výsledky detekce srovnajte s výsledky již implementovaných algoritmů. Diskutujte výsledky zvolené implementace s ohledem na detekční účinnost a rychlost výpočtu.

### DOPORUČENÁ LITERATURA:

- [1] YANG, M. H., KRIEGMAN D. J., AHUJA, N. Detecting faces in images: a survey. IEEE Transactions on Pattern analysis and Machine Intelligence. 2002, vol. 24, no. 1, p. 34-58.
- [2] CHEN, Y. J., LIN, Y., C. Simple face-detection algorithm based on minimum facial features. In Proceedings of the 33rd Annual Conference of the IEEE Industrial Electronic Society IECON 2007. Taipei, 2007, p. 455-460.

**Termín zadání:** 7.2.2011

**Termín odevzdání:** 27.5.2011

**Vedoucí práce:** Ing. Martin Mézl

**prof. Dr. Ing. Zbyněk Raida**  
*Předseda oborové rady*

### UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## **ABSTRAKT**

V této práci je prezentován přehled metod detekce obličeje v obraze a jsou vysvětleny základní principy klasifikace obrazu a jeho částí. Klíčovou částí práce je představení detektoru Viola-Jones a popis jeho implementace v jazyce Matlab. Detektor Viola-Jones je v praxi nejpoužívanější metoda pro detekci obličeje v obraze, což bylo důvodem pro detailní rozbor metody a následnou realizaci. Detektor je popsán teoreticky, rozebrány jsou základní kroky algoritmu a je zdokumentován trénovací algoritmus. Na základě teoretického rozboru byl detektor implementován v jazyce Matlab. Vlastnosti detektoru byly objektivně vyhodnoceny a porovnány s dalšími dvěma implementacemi detektoru Viola-Jones.

## **KLÍČOVÁ SLOVA**

Detekce obličeje v obraze, Viola-Jones Detektor, Testování detektoru Viola-jones, Adaboost

## **ABSTRACT**

This paper presents an overview of face detection methods. Keywords and basic principles of classification of images and its parts are explained. Significant part of this paper is occupied with presentation of Viola-Jones detector and its implementation in Matlab. Detector Viola-Jones ranks among the most used methods for face detection in practice, which was the reason for detailed analysis and subsequent implementation. Detector is theoretically described, basic steps of algorithm and training algorithm are discussed. Based on theoretical analysis, detector is implemented in Matlab. Properties of implemented detector are objectively evaluated and compared with of two different implementations.

## **KEYWORDS**

Face detection in images, Viola-Jones detector, Testing of Viola-Jones detector, Adaboost

Malach T., *Detekce obličeje v obraze*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií. Ústav radioelektroniky, 2011. 64 s, 2s příloh. Bakalářská práce. Vedoucí práce: ing.Martin Mézl

## PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma Detekce obličeje v obraze jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne .....

.....

(podpis autora)

## PODĚKOVÁNÍ

Děkuji vedoucímu bakalářské práce Ing. Martinu Mězlovi za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé bakalářské práce. Dále děkuji Ing. Jiřímu Přinosilovi Ph.D. za konzultace, jeho cenné zkušenosti a za poskytnutí natrénovaných dat. Děkuji také Ing. Petrovi Bambuchovi za pomoc při testování algoritmů. Děkuji také všem figurantům v testovacích množinách.

V Brně dne .....

.....

(podpis autora)

# OBSAH

<b>Seznam obrázků</b>	<b>vi</b>
<b>Seznam tabulek</b>	<b>viii</b>
<b>1 Úvod</b>	<b>9</b>
<b>2 Detekce obličeje v obraze</b>	<b>10</b>
2.1 Obecný algoritmus pro detekci obličeje .....	10
2.2 Hodnocení metod detekce obličeje .....	11
<b>3 Metody detekce obličeje v obraze</b>	<b>14</b>
3.1 Metody založené na znalosti .....	16
3.2 Metody založené na invariantních rysech .....	17
3.2.1 Textury obličeje .....	20
3.2.2 Barva kůže .....	20
3.2.3 Násobné rysy .....	22
3.3 Metody porovnávání se vzorem .....	22
3.4 Metody založené na vzhledu .....	24
3.4.1 Eigenfaces .....	24
3.4.2 Distribution-based methods .....	26
3.4.3 Neural networks –Neuronové sítě .....	28
3.4.4 Support vector machine .....	32
3.4.5 Další appearance-based methods .....	34
3.5 Výsledky metod .....	34
<b>4 Detektor Viola-jones a jeho implementace</b>	<b>36</b>
4.1 Příznaky – Haarovy filtry .....	36
4.2 Integrální obraz .....	37
4.3 Klasifikátor .....	38
4.4 Vývojový diagram detektoru .....	40
4.5 Trénování algoritmu - Adaboost .....	41
4.5.1 Trénování kaskádního klasifikátoru .....	43
4.6 Implementace detektoru .....	45

4.7	Funkce a funkční bloky algoritmu .....	45
4.7.1	Hlavní vlákno.....	45
4.7.2	Integrální obraz .....	45
4.7.3	Výpočet příznaků a slabé klasifikátory.....	46
4.7.4	Silný klasifikátor.....	47
4.7.5	Shluková analýza .....	48
4.7.6	Vývojový diagram implementace detektoru Viola - Jones.....	50
4.8	Uživatelské rozhraní .....	51
<b>5</b>	<b>Testování Detektoru</b>	<b>52</b>
5.1	Vstupní předpoklady.....	52
5.2	Testování.....	53
5.3	Vlastnosti testovací databáze .....	54
5.4	Databáze pro testování odolnosti proti náklonu hlavy.....	54
5.5	Výsledky testů.....	55
5.5.1	Výsledky testu na databázi .....	55
5.5.2	Diskuze výsledků .....	56
5.5.3	Výsledky testu odolnosti detektorů proti naklonění hlavy .....	57
5.6	Výpočetní čas a efektivita.....	58
<b>6</b>	<b>Závěr</b>	<b>59</b>
<b>7</b>	<b>Literatura</b>	<b>60</b>
<b>8</b>	<b>Seznam Zkratk a definic</b>	<b>63</b>
<b>9</b>	<b>Glosář</b>	<b>63</b>
	<b>Seznam příloh</b>	<b>65</b>

# SEZNAM OBRÁZKŮ

Obr. 2.1	Blokové schéma detekce obličeje. ....	11
Obr. 2.2	Ilustrativní ROC křivka. Převzato z [13]. ....	12
Obr. 2.3	ROC křivka detektoru Viola-Jones. Na vodorovné ose je vyneseno false positive rate v rozsahu $10^{-3}$ , na svislé ose je uveden detekční poměr. Převzato z [1]. ....	12
Obr. 3.1	a) Vstupní obraz ve kterém jsou vyznačeny profily, pro ilustraci je použit jednoduchý obraz s téměř konstantním pozadím, b) vertikální profil vstupního obrazu a), c) horizontální profil vstupního obrazu a).....	17
Obr. 3.2	Model obličeje podle Yow a Cippola, PFG, pro detekci obličeje v různých úhlu a dílčí části PFG, Hpairs a Vpairs. Obrázek převzat z [17]. ....	18
Obr. 3.3	Modely základních obličejových rysů. Sestávají vždy ze dvou hran. Obrázek převzat z [17]. ....	18
Obr. 3.4	Obr. Převzat z [17] Proces sestavování vertikálních a horizontálních párů, následně konstrukce PFG a podle modelu obličeje je sestaven kandidát na obličej.....	19
Obr. 3.5	Baesova síť slouží pro určení pravděpodobnosti, že PFG je sestaveno ze správně detekovaných rysů. Převzato z [17].....	19
Obr. 3.6	Hodnoty chrominančních komponent pro bravu kůže. Převzato z [10]. ....	20
Obr. 3.7	Vývojový diagram algoritmu, který prezentovali Chen a Lin. Převzato z [9]. ....	21
Obr. 3.8	Blokové schéma algoritmu. Převzato z [18]. ....	23
Obr. 3.9	Vzor pro oči vlevo a vzor pro obličej vpravo. Převzato z [18]. ....	23
Obr. 3.10	Vytvoření vektoru $\Gamma$ . Převzato z [12]. ....	24
Obr. 3.11	Množina obrazů $\Phi$ od kterých je odečten průměrný vektor. Převzato z <a href="http://www.cs.princeton.edu/~cdecoro/eigenfaces/">http://www.cs.princeton.edu/~cdecoro/eigenfaces/</a> . ....	25
Obr. 3.12	V horní řadě postupně zleva, množina obrazů tváře v trénovací množině, reprezentace tváří v mnohadimenzionálním prosotru, clusterované shluky reprezentací obrazů tváří a nejvíce pravděpodobné tváře, tj. ty, které jsou v těžistích. V druhé řadě postupně zleva, množina obrazů okolí v trénovací množině, jejich reprezentace v mnohadimenzionálním prosotru společně s clustery tváří, clusterované reprezentace tváří i okolí, těžisté clusterů obrazů okolí. Převzato z [20]. ....	26
Obr. 3.13	Obrázek a) znázorňuje postupně vytvoření trénovací množiny obrazů tváře, postupně, výběr obličeje, maska, použitá maska, pro snížení dimenzionality prosotru a vyjmutí irelevantního okolí tváře. b) zobrazuje vstupní obraz rozdělený na překrývající se subobrazy. c) vývojový diagram algoritmu,	



	který prezentovali Sung a Poggio[20]. Převzato z [20]. ....	27
Obr. 3.14	Blokové schéma jednoho neuronu. Převzato z [23] a modifikováno. ....	28
Obr. 3.15	Schéma neuronové sítě. ....	29
Obr. 3.16	Postup algoritmu, který prezentovali Rowley, Baluja a Kanade. Převzato z [22]. ....	30
Obr. 3.17	Předzpracování vstupního obrazu. Převzato z [22]. ....	31
Obr. 3.18	Zobrazení dvou množin bodů v rovině. Převzato z [26] a upraveno. ....	32
Obr. 3.19	Možné oddělení dvou skupin bodů pomocí různých lineárních klasifikátorů. Převzato z [24] a modifikováno. ....	33
Obr. 3.20	Zobrazení dvou množin dat ve vícedimenzionálním prostoru a nalezení oddělující roviny. Převzato z [26] a upraveno. ....	33
Obr.4.1	Haarovy filtry. A horizontální (hranový) filtr, B vertikální (hranový) filtr, C trojnásobný (čárový) filtr, D čtyřnásobný (diagonální) filtr. Převzato z [1]. ....	36
Obr. 4.2	Výpočet jasových hodnot v obdélníku D. Převzato z [1]. ....	37
Obr. 4.3	Schéma klasifikace.....	38
Obr. 4 4	Obecný vývojový digram detektoru Viola-Jones. ....	40
Obr. 4.5	Vývojový diagram algoritmu Adaboost pro trénování monolitického klasifikátoru .....	42
Obr. 4.6	Vývojový algoritmus trénování kaskádního klasifikátoru .....	44
Obr. 5.1	Příklad obrázku z testovací databáze .....	54
Obr. 5.2	Specifikace os okolo kterých, se otáčí hlava na obrazech v databázi pro testování odolnosti proti naklonění hlavy. ....	55
Obr. 5.3	Příklad výsledku implementace detektoru TM .....	56
Obr. 5.4	Výsledek testu natočení hlavy okolo vertikální osy detektoru Vahida Kazemiho .....	58

# SEZNAM TABULEK

Tabulka 1	Přehled metod detekce obličeje .....	15
Tabulka 2	Přehled výsledků metod pro detekci obličeje. ....	35
Tabulka 3	Výsledky testování.....	55
Tabulka 4	Mezní úhly naklonění hlavy okolo horizontální oční osy.....	57

# 1 ÚVOD

Zpracování obrazu dnes patří k dynamicky se rozvíjejícím oborům. Pokroky v tomto odvětví je možné vidět jak v běžném životě, tak i v průmyslu. Na základě zpracování obrazu a extrakce informace z obrazu můžeme provádět kontrolu výrobků na výrobní lince, kontrolovat kompletnost sad výrobků v jednotlivých baleních, ale také detekovat projíždějící auta a zaznamenávat jejich poznávací značky. Zpracování obrazu hraje také velmi významnou roli v lékařské diagnostice. Pro každou z těchto aplikací je použit velmi úzce zaměřený algoritmus. Algoritmus je zpravidla hlavní faktor, určující výsledky systému zpracovávající obraz. To je důvodem pro stálé zdokonalování a vývoj nových algoritmů.

Zpracování obrazu za účelem extrakce biometrických dat z obrazu, je nazývána face recognition neboli rozpoznávání tváře. S prvními algoritmy pro rozpoznávání tváře se dnes již můžeme setkat v některých elektronických zařízeních, například ve fotoaparátech. V budoucnosti by rozpoznávání tváře mohlo doplnit nebo dokonce i nahradit dnešní způsoby pro identifikaci osob. To by znamenalo, otevírání dveří, přihlášení k PC nebo platební operace na základě tváře, proto je rozpoznávání tváře dnes velmi aktuálním tématem.

Jednou z částí velmi rozsáhlé problematiky rozpoznávání tváře je detekce tváře, téma této práce.

V první části této práce jsou uvedeny obecné základy metod pro detekci obličeje. Následně jsou tyto metody analyzovány a popsány. Metod pro detekci obličeje v obraze již bylo prezentováno mnoho. V této práci budou zdůrazněny hlavně ty algoritmy detekce obličeje, které byly významnými milníky ve vývoji a také ty, které jsou významné z hlediska jejich dalšího rozvoje.

Prvořadým cílem práce je stanovení základních teoretických znalostí v oblasti detekce obličeje v obraze a metod, které jsou používány pro detekci obličeje v obraze. Druhý významný úkol spočívá v prezentaci detektoru Viola Jones a jeho implementaci v prostředí Matlab. Ten se jeví z hlediska svých vlastností jako optimální detektor obličeje v obraze.

V Kapitole 2 je popsán obecný postup algoritmu detekce obličeje. Jsou diskutovány způsoby hodnocení jednotlivých metod a hlavní faktory ovlivňující výsledky metod.

V kapitole 3 jsou představeny metody pro detekci obličeje v obraze a jejich kategorizace. Je zde také uveden přehled výsledků jednotlivých metod.

V kapitole 4 je teoreticky představen detektor Viola-Jones a trénovací algoritmus Adaboost. Na základě rozboru algoritmu byl vytvořen funkční algoritmus a jeho implementace je detailně popsána. Detektor Viola-Jones byl vybrán pro implementaci a detailní studium, protože z hlediska jeho výkonnosti a vlastností se jeví jako optimální algoritmus pro reálné nasazení v praxi.

V kapitole 5 jsou shrnuty výsledky testování detektoru a porovnání s ostatními implementacemi detektoru Viola-Jones.

## 2 DETEKCE OBLIČEJE V OBRAZE

Detekce obličeje je proces, jehož cílem je rozhodnout zda v libovolném vstupním obraze se nacházejí obličeje, pokud ano, výstupem procesu je pozice obličeje v obraze.

### 2.1 Obecný algoritmus pro detekci obličeje

Každá metoda pro rozpoznávání objektů v obraze sestává ze dvou základních částí.

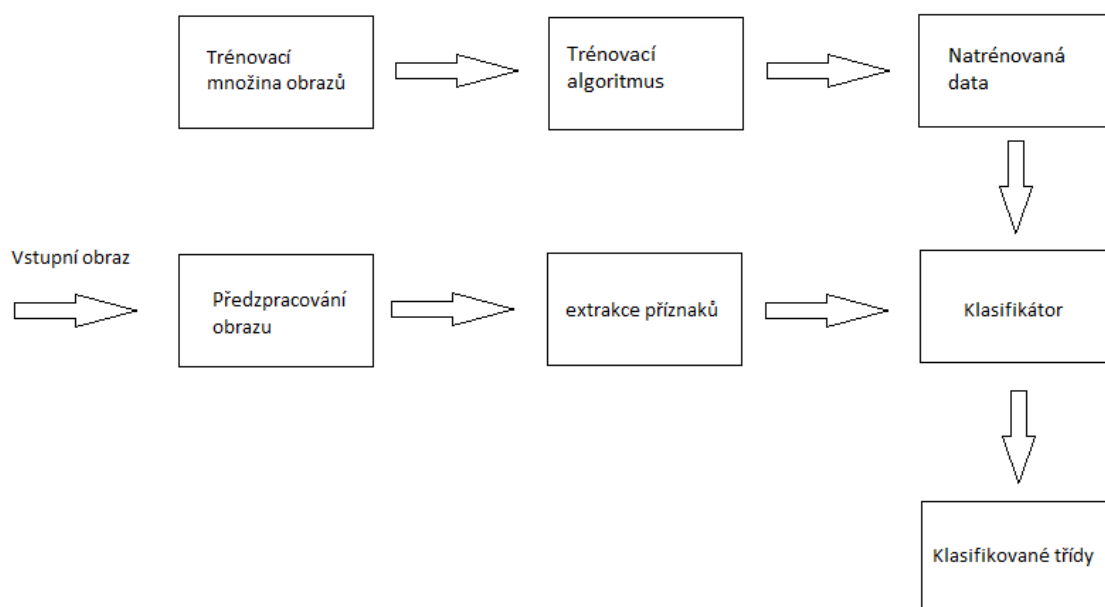
První základní částí každého algoritmu pro detekci obličeje v obraze je proces extrakce tzv. příznaků, viz. obrázek 2.1. Příznakem může být cokoliv, co charakterizuje objekt a jeho vlastnosti. Příkladem příznaku může být barva objektu nebo vzdálenost jeho reprezentace v mnohorozměrném prostoru od referenčního bodu.

Proces extrakce příznaků má zajistit dostatečné množství relevantních informací, které jsou následně posouzeny druhým základním celkem – klasifikátorem.

Klasifikátor je algoritmus, který je schopen rozčlenit vstupní data do předem definovaných skupin [13]. Rozčlenění vstupních dat je provedeno na základě nějakého rozhodovacího pravidla. Toto rozhodovací pravidlo zásadně ovlivňuje výsledky klasifikačního procesu, proto je velmi pečlivě vybíráno. Jak bude ukázáno v následující kapitole, rozhodovací pravidla mohou mít mnoho podob. Rozhodovací pravidlo, může být mezní vzdálenost dvou bodů, nalezení jistých charakteristických prvků nebo definovaný rozptyl chrominanční komponenty obrazových bodů.

V případě detekce obličeje se jedná o binární klasifikaci. Rozhoduje se o tom, zda-li se v daném obraze obličej nachází nebo ne. Zde je zcela evidentní rozdíl oproti rozpoznávání tváře, kde každá skupina obsahuje jeden obličej a takovýchto skupin může být velké množství.

Pokročilé klasifikátory jsou učící se algoritmy. Tyto algoritmy musí být trénovány. Během trénování jsou klasifikátoru předkládány data s informací, do které skupiny tato data patří, viz. obrázek 2.1. Klasifikátor si sám tímto způsobem stanoví rozhodovací pravidlo, které je proměnné a závisí na trénovací množině dat.



Obr. 2.1 Blokové schéma detekce obličeje.

## 2.2 Hodnocení metod detekce obličeje

Základním kriteriem pro hodnocení těchto metod je detection rate – detekční poměr, (detection rate, true positive rate, TPR). Detekční poměr je poměr<sup>1</sup> počtu správně detekovaných obličejů algoritmem a počtu obličejů určených člověkem [14], označuje se někdy jako TPR (true positives rate).

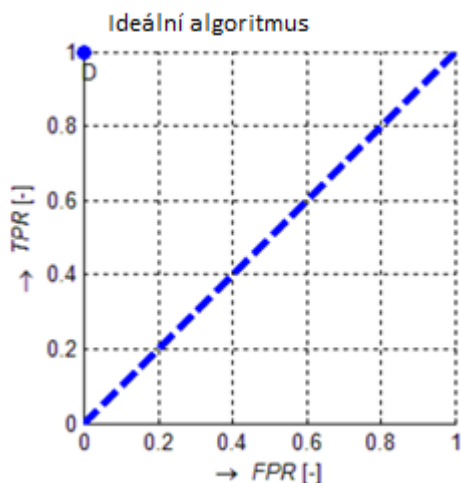
Algoritmy mohou udělat dva typy chyb, false negatives a false positives. V terminologii detekce tváře je chybná klasifikace vyjádřena slovem false, druhé slovo vyjadřuje typ chyby. False positives je taková chyba, kdy jsou jako obličej označeny oblasti obrazu, kde se tvář nevyskytuje, označuje se jako halucinace. False negatives je chyba, kdy existující tváře v obrazu, nejsou označeny, označuje se jako přehlédnutí.

False positives a detekční poměr společně velmi úzce souvisejí. S rostoucím detekčním poměrem roste i počet false positives a naopak [14]. Vztah mezi false positives a detekčním poměrem je společnou vlastností pro všechny metody detekce obličeje. V praxi je třeba stanovit kompromis mezi detekčním poměrem a počtem false positives. Z výše uvedeného vyplývá, že při hodnocení metod je nezbytné uvážit jak detekční poměr, tak počet false positives.

Vztah mezi detekčním poměrem (TPR) a počtem false positives (false positive rate, FPR) se obvykle vynáší do grafu, který je označován jako ROC křivka (Receiver operating characteristic). Ilustrační ROC křivka je na obrázku 2.2. Vztah mezi detekčním poměrem a false positives rate klasifikátoru je možné reprezentovat bodem na křivce ROC[13]. Modrá křivka na obrázku 2.2 reprezentuje ty algoritmy, které

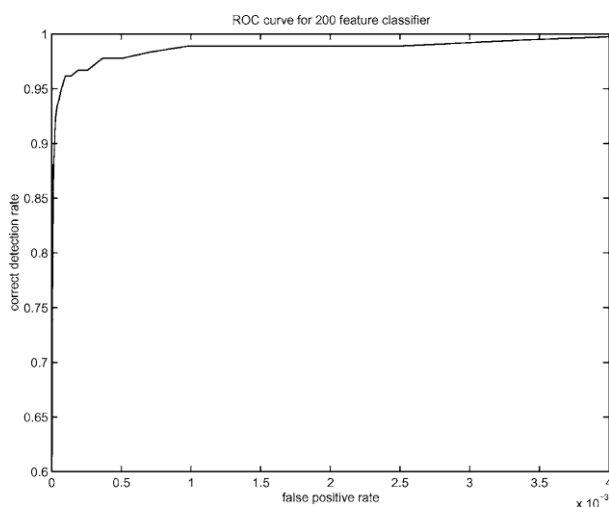
<sup>1</sup> Poměr je vyjádřen v procentech.

pracují zcela náhodně tzn., že je stejný počet false positives jako správně detekovaných obličejů. V levém horním rohu je reprezentován ideální algoritmus se 100% detekčním poměrem a nulovou chybovostí. Nad modrou osou se nacházejí všechny algoritmy, které pracují lépe než náhodný klasifikátor.



Obr. 2.2 Ilustrativní ROC křivka. Převzato z [13].

Reálná ROC křivka je zobrazena na obrázku 2.3.



Obr. 2.3 ROC křivka detektoru Viola-Jones. Na vodorovné ose je vynesena false positive rate v rozsahu  $10^{-3}$ , na svislé ose je uveden detekční poměr. Převzato z [1].

Při hodnocení metod je nutné vzít ohled i na další faktory, které zásadně ovlivňují výsledky testování. Tyto faktory charakterizují metody z hlediska jejich robustnosti. Mezi tyto faktory patří [14]:

- Úhel natočení obličeje v obraze. Některé metody nejsou schopné detekovat natočené hlavy, jiné jsou toho schopny do určité míry natočení.
- Přítomnost nebo absence některých rysů obličeje. Příkladem mohou být brýle, vlasy nebo vousy.

- Výraz tváře. Při změně výrazu může dojít k potlačení některých rysů.
- Překrytí tváře jiným objektem. Velmi často se vyskytující se případ v reálných obrazech. Tento faktor má zásadní vliv na výsledek většiny metod.
- Kvalita obrazu. Kvalitou obrazu rozumíme míru šumu v obraze, kontrast obrazu, celkové jasové podmínky, případné ostré stíny.

Dalším parametrem pro hodnocení je výpočetní náročnost algoritmu, která je úměrná výpočetnímu času. Tento údaj u mnoha prezentovaných metod chybí. V poslední době se ovšem výpočetní doby pohybují v takových mezích, že je možné použít detektorů pro detekci obličeje ve videosekvencích s rychlostí používanou v kamerových systémech (až 25 snímků za sekundu).

Pro komplexní hodnocení metod je tedy nutné uvést a specifikovat všechny výše uvedené charakteristiky. Z povahy problematiky je již nyní zřejmé, že není možné vyhodnotit nejlepší ani nejhorší metodu, protože každá metoda má své výhody a nevýhody. V závěru následující kapitoly budou proto charakteristiky metod pouze shrnuty.

### 3 METODY DETEKCE OBLIČEJE V OBRAZE

V této kapitole budou prezentovány metody pro detekci obličeje. Yang, Kriegman a Ahuja v roce 2002 v jejich rešeršní práci [14] rozdělili metody detekce obličeje do čtyř skupin <sup>2</sup>:

- Knowledge-based methods – metody založené na znalosti, znalostní metody.
- Feature invariant methods – metody založené na invariantních rysech.
- Template matching – metody porovnávání se vzorem.
- Appearance-based methods – metody založené na vzhledu.

Metody založené na znalosti obsahují informaci, z jakých částí se skládá lidská tvář a jaké jsou mezi jednotlivými částmi obličeje vztahy. Algoritmy detekují různé části obličeje a dále zkoumají vztahy mezi nimi, za účelem rozhodnutí, jedná-li se o tvář.

Metody založené na invariantních rysech detekují různé obličejové rysy a na základě jejich vlastností a vztahů je rozhodnuto o přítomnosti tváře. Obličejové rysy jsou vybírány tak, aby bylo možné je detekovat i když se mění osvětlení nebo natočení obličeje.

Metody porovnávání se šablonou porovnávají vzory obličeje a obličejových rysů. Korelace mezi vstupním obrazem a daným vzorem je měřítkem pro posouzení přítomnosti obličeje.

Metody založené na vzhledu porovnávají vstupní obraz s modelem obličeje, který je vytvořen během trénování klasifikátoru [33]. Model může mít podobu rozdělovacích funkcí nebo distribučního modelu.

Zástupci jednotlivých metod a jejich příslušnost ke skupinám metod je uvedena v tabulce.

---

<sup>2</sup> V češtině dosud nebyly ustáleny názvy metod, je proto možné se setkat s jinými názvy.



Metody založené na znalosti	Multiresolution rule-based methods	Human Face Detection in Complex Background [16]
		Rule-based face detection in frontal views [15]
Metody založené na invariantních rysech		Framework for Perceptual Grouping of Features for Human Face Detection [17]
		Fast face detection via morphology-based pre-processing [19]
	Textury	Face-texture model based on SGLD and it's application in face detection in a color scene []
	Barva kůže	Simple Face-detection Algorithm Based on Minimum Facial Features [9]
	Násobné rysy	Face Detection Using Color Thresholding, and Eigenimage Template Matching [10]
Metody porovnávání se vzorem	Predefined templates – předdefinované vzory	Face detection through template matching and color segmentation [18]
Metody založené na vzhledu	Eigenfaces	Eigenfaces for Face Detection/Recognition [12]
	Distribution-based methods	Example-Based Learning for View-Based Human Face Detection [20]
		Probabilistic Visual Learning for object Recognition [21]
	Neural Networks – Neuronové sítě	Neural Network-Based Face Detection [22]
	Support vector machine	Training support vector machines: An application for face detection [24]
	Boosting	Viola-Jones detektor [1]
	Další appearance-based methods	A SNoW face detektor [34]
		Hidden Markov Models [35]

Tabulka 1 Přehled metod detekce obličeje

Tento přehled vychází z přehledu uvedeného ve [14].

Toto rozdělení je jen přibližné, protože většina metod je na pomezí skupin nebo má takový charakter, že není možné je jednoznačně zařadit.

### 3.1 Metody založené na znalosti

Metody založené na znalosti využívají lidskou znalost o vzhledu obličeje. Za znalost obličeje můžeme považovat rysy vyskytující se v obličeji (oči, ústa, nos atd.) a jejich vzájemné vztahy. Algoritmy mohou používat různé množiny rysů pro detekci. Klasifikace zkoumané oblasti obrazu je založena na předdefinovaném souboru rozhodovacích pravidel. Soubor pravidel obvykle představují prostorové vztahy rysů a jejich počet.

Obtíží těchto metod je specifikace souboru rozhodovacích pravidel, tedy přenesení lidské znalosti o vzhledu obličeje do algoritmu. Bylo zjištěno, že pokud soubor pravidel obsahuje příliš detailní informace, nemusí odpovídat všem tvářím a výsledkem je nízký detekční poměr. Naopak je-li soubor pravidel příliš obecný, dochází k chybám false positives.

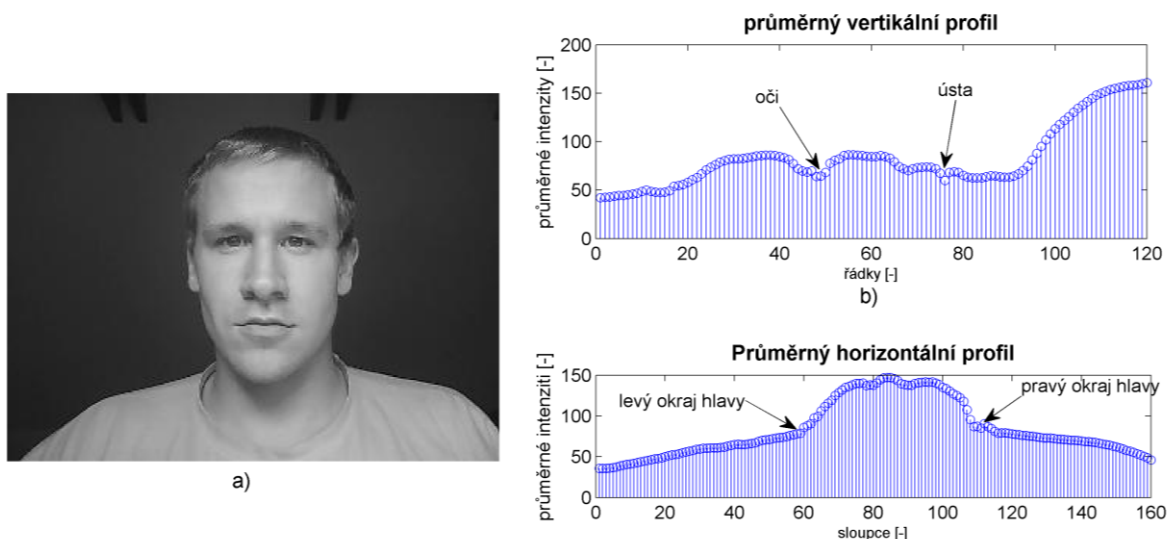
Již z povahy této skupiny metod je zřejmé, že není možné detekovat obličeje, které nejsou obsaženy ve vstupním obraze zepředu. Rozšíření algoritmu na detekci tváří z různých úhlů, by bylo velmi náročné a navíc by to negativně ovlivnilo výpočetní dobu.

Algoritmus, který je založen na lidské znalosti, publikovali Yang a Huang [16]. Vytvořili třístupňový algoritmus, který zpracovával černobílý obraz ve třech různých rozlišeních. Při každé úrovni rozlišení jsou extrahovány jiné rysy, pro které je definován jistý soubor rozhodovacích pravidel a vztahů, na jehož základě probíhá klasifikace oblasti. Různé velikosti jsou realizovány pomocí podvzorkování a následného průměrování.

Je třeba poukázat na výhodu této metody. Postup od nízkého k většímu rozlišení, je strategický, neboť jednotlivé procesy mají adekvátní množství informací, které zpracovávají. Tím pádem dochází úspoře času při výpočtu.

Na tento algoritmus navázali Kotropoulos a Pitas [15], v jejich metodě bylo využito principu metody, kterou prezentovali Yang a Huang. Zaměřili se na zvýšení účinnosti algoritmu. Do algoritmu byl přidán proces předzpracování, jehož úlohou je odhad velikosti oblasti obličeje a zároveň jeho lokalizace v obraze. Tímto procesem je nahrazen iterační proces, který algoritmu, publikovaném v [16], procházel celý obraz v různých měřítkách za účelem nalezení kandidátních oblastí na obličeje. V [15] byl tento iterační proces nahrazen krokem, ve kterém je zpracováván průměrný horizontální a vertikální profil vstupního obrazu.

Příklad profilů je vidět na obrázku 3.1 b) i c), některé rysy jasně viditelné. Algoritmus vyhledá ty řádky resp. sloupce, ve kterých se významné obličejové rysy nacházejí. Podle polohy těchto rysů je definována oblast, ve které je očekáván obličej. Oblast je rozdělena na podoblasti, mezi kterými musí platit jisté vztahy [15], aby nebyla celá oblast vyloučena jako false positive. V oblasti jsou přesně detekovány potřebné rysy. Následně proběhne verifikace oblasti, kritériem jsou vztahy obličejových rysů, detailně pospané v [15].



Obr. 3.1 a) Vstupní obraz ve kterém jsou vyznačeny profily, pro ilustraci je použit jednoduchý obraz s téměř konstantním pozadím, b) vertikální profil vstupního obrazu a), c) horizontální profil vstupního obrazu a).

Tato metoda předpokládá výskyt maximálně jedné tváře v obraze. Detekce oblastí obličeje z průměrných profilů by zřejmě nefungovala při výskytu mnoha tváří. Tváře musí být také vzhledem k obrazu dostatečně velké, aby se obličejová část v průměrných profilech projevila. Zásadní vliv na úspěšnost metody má osvětlení a kontrast obrazu. Jak je uvedeno ve [14], jsou metody založené na znalosti určeny spíše k lokalizaci obličeje.

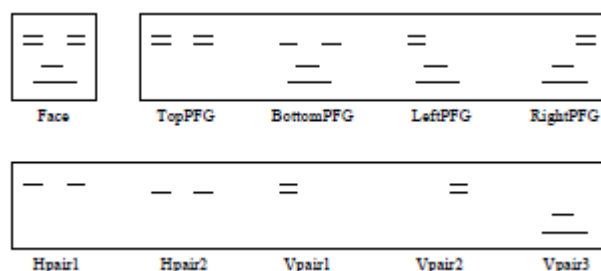
### 3.2 Metody založené na invariantních rysech

Metody založené na invariantních neboli neměnných rysech vychází z představy, že musí existovat charakteristické prvky nebo vlastnosti, které jsou stejné pro všechny tváře. Výzkum těchto metod je motivován faktem, že člověk je schopen rozeznat lidskou tvář velmi rychle a spolehlivě. Musí tedy existovat nějaké prvky tváře nebo vlastnosti, podle kterých lidé rozeznávají tváře. Hledání těchto prvků nebo vlastností a jejich využití pro detekci obličeje je pojičkem mezi metodami založených na invariantních rysech.

Společným problémem metod založených na invariantních rysech je citlivost na šum v obraze a velký vliv osvětlení. Obličejové rysy jsou extrahovány z obrazu obvykle pomocí hranových detektorů. Při zvýšeném šumu nebo špatném osvětlení mohou být hrany potlačené. Extrémním případem mohou být stíny, kdy dojde k detekci hrany, která vůbec obličejí nenáleží.

Yow a Cippola [17] představili algoritmus, který důmyslným způsobem detekuje obličejové rysy, ty sdruží do skupin a podle pravděpodobnostního schématu – Baesovské sítě, rozhodne, zda tyto rysy v daném uspořádání tvoří obličej.

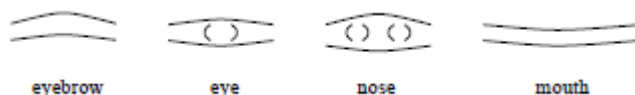
Yow a Cippola modelují obličej jako oblast s šesti výraznými rysy. Tyto rysy (obočí, oči, ústa atp.) jsou sdruženy do čtyř skupin, nazvaných PFG, (PFG – partial face groups) po čtyřech rysech. Rysy jsou uspořádány tak, že skupiny představují významné obličejové rysy viditelné z různých úhlů, což umožní detekci i při různých natočeních hlavy. Pro efektivnější konstrukci Baesovské sítě jsou PFG rozděleny na další dílčí části po dvou rysech, tzv. Hpairs a Vpairs, viz. obrázek 3.2, dolní řádek.



Obr. 3.2 Model obličeje podle Yow a Cippola, PFG, pro detekci obličeje v různých úhlu a dílčí části PFG, Hpairs a Vpairs. Obrázek převzat z [17].

Trénování tohoto algoritmu spočívá v extrakci příznaků z trénovací množiny obrazů.

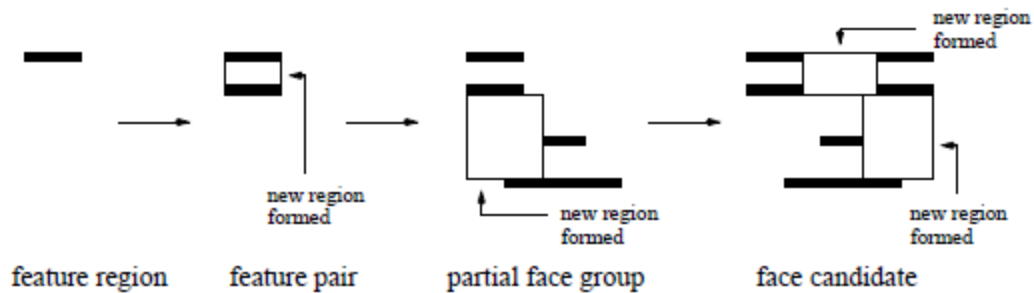
Prvním krokem při detekci obličeje je nalezení oblastí s možným výskytem tváře, ve vyhlazeném vstupním obraze. V obraze jsou vyhledána lokální maxima, reprezentující světlé oblasti, představující body zájmu. V okolí těchto lokálních maxim je provedena detekce hran. Z hran jsou mohou být sestaveny jednotlivé rysy, viz. obrázek 3.3. Hrany, ze kterých je složen rys musí být přibližně rovnoběžné a každá musí mít opačný gradient jasu. Z hran splňujících tyto podmínky jsou sestaveny rysy. Z hran rysů se extrahují další informace – příznaky rysu, např. délka hrany, šířka hrany a rozdíl jasu před a za hranou atd. [17].



Obr. 3.3 Modely základních obličejových rysů. Sestávají vždy ze dvou hran. Obrázek převzat z [17].

Všechny vektory příznaků rysů jsou zobrazeny v příznakovém prostoru. Pokud je Mahalanobisova vzdálenost v příznakovém prostoru mezi některým z vektorů s příznaky rysů z trénovací množiny a vektorem příznaků potenciálního rysu menší než nastavený práh, jsou tyto dvě hrany prohlášeny za konkrétní rys.

Takto jsou v obraze detekovány všechny možné rysy na různých pozicích v obraze. Tyto rysy jsou kombinovány tak, že vytvoří několik PFG viz. obrázek 3.4. Ze sestavených PFG jsou extrahovány příznaky charakterizující každé PFG. PFG musí splňovat určitá pravidla. Tyto pravidla jsou vyjádřena hodnotami příznaků z natrénovaných PFG. Vytvořené PFG jsou verifikovány. Verifikace se uskuteční na základě prahování Mahalanobisovy vzdálenosti mezi vektorem příznaků vytvořeného PFG a vektorem příznaků PFG z trénovací množiny. Verifikace příznaků sníží počet false positives.

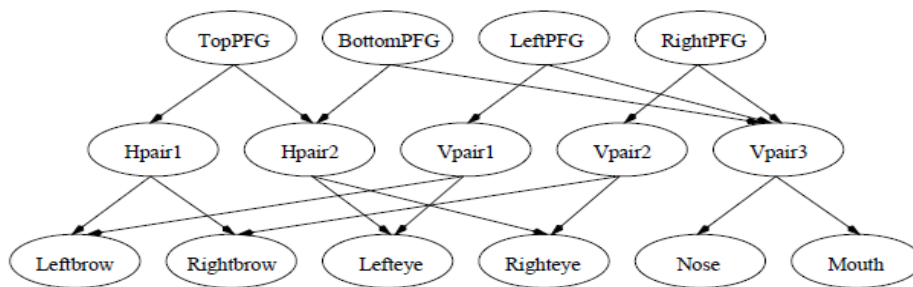


Obr. 3.4 Obr. Převzat z [17] Proces sestavování vertikálních a horizontálních párů, následně konstrukce PFG a podle modelu obličeje je sestaven kandidát na obličej.

Mezi verifikovanými PFG se mohou stále vyskytovat i PFG sestavené z chybně verifikovaných rysů. Takovéto PFG by zvyšovali počet false positives. Eliminace takových PFG je provedena pomocí Baesovy pravděpodobnostní sítě. Každému obličejovému rysu je přiřazena pravděpodobnost  $P_i$ , že byl správně detekován. Pravděpodobnost  $P_i$  určuje rovnice 3.1:

$$P_i = 1 - \frac{M_{ij}}{\tau_j}, \quad (3.1)$$

kde  $M_{ij}$  je Mahalanobisova vzdálenost mezi vektorem reprezentujícím daný rys  $X_i$  a vektorem rysu z trénovací množiny,  $\tau_j$  je prahová vzdálenost v příznakovém prostoru mezi vektory. Správně detekovaný rys má vyšší pravděpodobnost  $P_i$ . Dále jsou všechny pravděpodobnosti  $P_i$  rysů z každého PFG použity pro konstrukci Baesovy sítě viz. obrázek 3.5. Díky ní je možné rychle vypočítat celkovou pravděpodobnost pro každé sestavené PFG.



Pokud dojde ke špatné detekci rysu, bude mít PFG, obsahující tento rys nižší

Obr. 3.5 Baesova síť slouží pro určení pravděpodobnosti, že PFG je sestaveno ze správně detekovaných rysů. Převzato z [17].

celkovou pravděpodobnost, než PFG se správně detekovanými rysy. PFG s nižší pravděpodobností je vyřazeno. PFG s nejvyšší pravděpodobností je prohlášen za obličej.

Tato metoda dle autorů má úspěšnost detekce 92%. Testování proběhlo na obrazech o rozměru 256x256 pixelů. Algoritmus je schopen správně detekovat i obličeje, které nejsou zobrazeny zepředu. Brýle, částečné zakrytí tváře nebo absence některých rysů nemají výrazný vliv na správnou detekci.

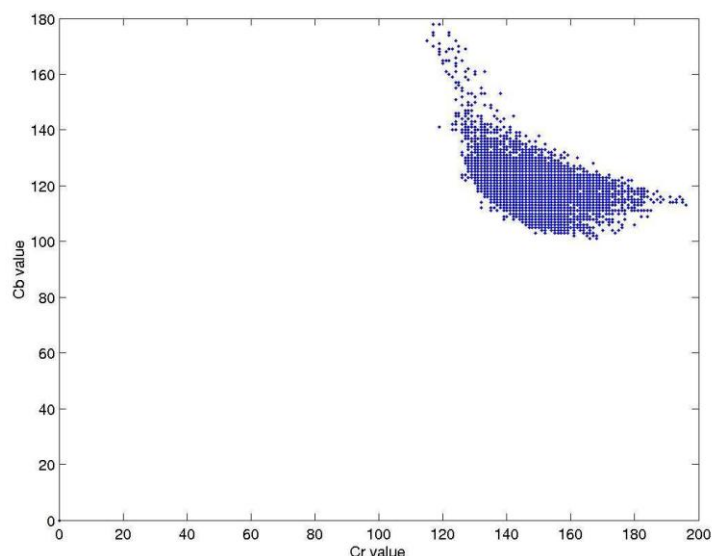
### 3.2.1 Textury obličej

Lidské tváře mají odlišnou vnitřní strukturu (texturu) od okolí i jiných předmětů. Toho bylo také využito pro vytvoření několika metod detekce obličej, popsanych níže.

### 3.2.2 Barva kůže

Detekci obličej je možné realizovat na základě znalosti rozložení barev v obraze. Metody detekující obličej podle barvy se opírají o hledání pixelů určité barvy v obraze – segmentace pixelů s barvou kůže (skin segmentation). Pixely jsou na základě předem definovaných podmínek prohlášeny za pixel s barvou kůže či okolí. Takto je možné v obraze rozeznat pixely s barvou kůže. Dalším krokem je vyloučení pixelů a skupin pixelů, které sice barvu kůže mají, ale nejsou obličej. K tomu je použito buďto maskových operátorů nebo detekce dalších obličejových rysů. Mnoho algoritmů používá různé barevné modely např. HSV, YCbCr, normalizovaný RGB a mnoho dalších.

V prostoru YCbCr je barva charakterizována pouze složkami Cb a Cr. Barva lidské kůže má ve složkách Cb a Cr jen malý rozptyl a obsahuje vysoké hodnoty komponent Cb a Cr viz. obrázek 3.6. Barvy kůže jak je vnímají lidé, jsou způsobeny především změnou jasu, ta je v prostoru YCbCr dána složkou Y[18]. Toho je využito pro detekci. Empiricky byly zjištěny prahové hodnoty složek Cb a Cr, které vymezují barvu kůže.



Obr. 3.6 Hodnoty chrominančních komponent pro barvu kůže. Převzato z [10].

Chen a Lin [10] prezentovali metodu, kde bylo využito normalizovaného prostoru RGB pro detekci kůže a prostoru HSV pro detekci vlasů. Tato kombinace zajišťuje nižší počet false positives.

Normalizovaný RGB prostor byl definován takto[10]:

$$r = \frac{R}{R + G + B}, \quad (3.2)$$

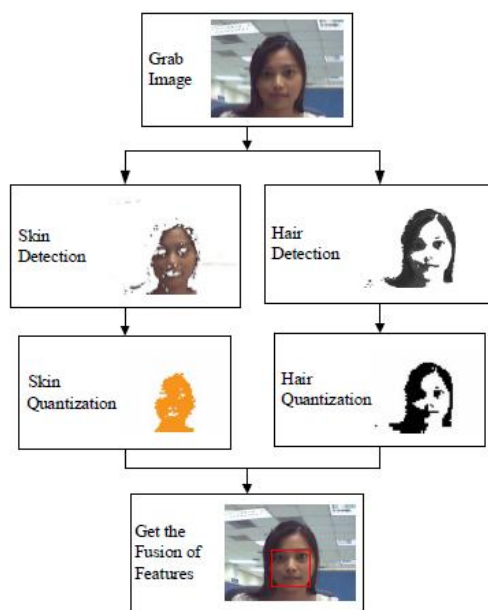
$$g = \frac{G}{R + G + B}, \quad (3.3)$$

Kde  $r$  a  $g$  jsou červená a zelená složka normalizovaného barevného modelu RGB a  $R, G, B$ , jsou hodnoty červené, zelené a modré složky barevného modelu RGB.

S využitím  $H$  (tón barvy) složky prostoru HSV a složek  $r$  a  $g$  normalizovaného prostoru RGB byla stanovena řada podmínek pro vyloučení pixelů s barvou neodpovídající barvě kůže viz. obrázek 3.7.( blok skin detection).

Detekce vlasů byla realizována použitím složky  $V$  (jas) prostoru HSV a složek  $R$  a  $G$  prostoru RGB. Jednoduchým stanovením prahů pro jednotlivé komponenty byla vybrána oblast vlasů viz. obrázek 3.7.( blok hair detection).

Kroky kvantizace vlasů a kůže viz obrázek 3.7 (bloky hair a skin quantization) spočívají ve výběr čtverců rozměru 5x5 pixelů. Čtvercům obsahující pixel vlasů nebo kůže je přiřazena hodnota 1 jiným 0 a tak je vytvořen binární obraz. Výběrem čtverců 5x5 pixelů dojde de facto k podvzorkování a tím i zrychlení dalších kroků algoritmu.



Obr. 3.7 Vývojový diagram algoritmu, který prezentovali Chen a Lin. Převzato z [9].

Posledním krokem algoritmu je porovnání, zda-li oblast pixelů s barvou kůže zasahuje nebo sousedí s oblastí pixelů s barvou vlasů. Oblasti musí sousedit tak, aby mezi nimi byl jistý geometrický vztah[10].

Chen a Lin v jejich práci uvádí úspěšnost detekce 92% při rychlosti 10 *fps*. Tato metoda je schopná detekovat obličeje zobrazené ve vstupním obraze i z boku.

### 3.2.3 Násobné rysy

Do této skupiny metod patří ty, které kombinují příznaky a rozhodovací pravidla použité v metodách uvedených v kapitolách 3.2.1 a 3.2.2. Obvykle je použito barevné segmentace k detekci kůže a změření významných obličejových rozměrů včetně ověření tvaru oblasti obličeje. Dále jsou detekovány detailní obličejové rysy. Detailní rysy slouží pro verifikaci oblastí, kde je očekáván obličej.

Marius, Pennathur a Rose představili v jejich práci [10] systém, který využívá transformace do prostoru YCbCr. Jsou nalezeny oblasti s barvou kůže, a pokud mají tyto oblasti definovaný rozměr a tvar, předpokládá se zde obličej. Obličej je předpokládán v těchto oblastech jsou verifikovány metodou eigenfaces, popsanou v kapitole 3.4.1

Systém uvedený v [10] využívá prahování chominančních komponent Cb a Cr pro segmentaci kůže. Obraz je převeden na binární tím způsobem, že pixely obsahující barvu kůže mají hodnotu 1 ostatní 0. Následně je použito několik průchodů jednoduchými filtry, které odstraní jednotlivé pixely a malé shluky pixelů, které byly špatně detekovány při prahování. Největší obtíží je nyní vyloučit ty oblasti, které mají barvu kůže, ale nejsou obličej. Toho je realizováno vyříznutím jednotlivých oblastí s barvou kůže (potencionálních tváří) do samostatných obrazů, avšak tyto oblasti musí spadat do jistého rozsahu velikostí.

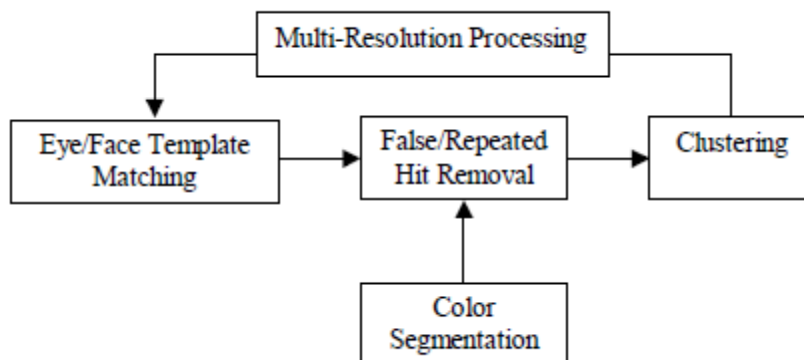
V těchto vyříznutých subobrazech jsou hledány eigenfaces. Jedná se vytvoření obrazu v subobrazech, pomocí předem vytvořených vzorů obličejů. Suboblast musí být možné sestavit jako lineární kombinaci těchto vzorů. Na základě metody eigenfaces je rozhodnuto o přítomnosti tváře v obraze.

## 3.3 Metody porovnávání se vzorem

Další možný přístup k problematice detekce tváře je template matching, neboli porovnávání se vzorem. Základem této metody je vytvoření souboru vzorů, které jsou následně porovnávány se vstupním obrazem, eventuálně jeho částmi a dle odezvy na tento vzor je rozhodnuto, jedná-li se o tvář nebo její části. Vzorem je reálný obraz. Metoda porovnávání se vzorem je velice jednoduše zobecnitelná a rozšiřitelná na detekci různých objektů.

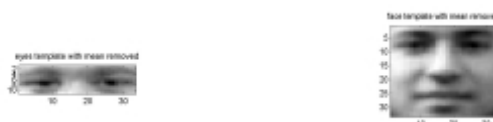
Ping, Weng a Lau [18] vytvořili kombinaci metody porovnávání se vzorem a segmentací barvy kůže algoritmus s poměrně vysokým detekčním poměrem a nízkým počtem false positives. Algoritmus se cyklicky opakuje s rozdílnou velikostí vstupního obrazu, viz. obrázek 3.8 (blok multi-resolution processing). Blokové schéma algoritmu je na obrázku 3.8. Nejdříve je provedena segmentace barvy kůže (blok color segmentation), poté je provedeno porovnání se vzorem (Eye/face template matching). Výsledky porovnávání se vzorem a barevné segmentace jsou vyhodnoceny v bloku False/repeated Hit removal. V bloku Clustering jsou shromážděny informace o detekovaných obličejích a násobně detekované obličejové jsou sloučeny. Algoritmus se opakuje pro jinou velikost vzoru.





Obr. 3.8 Blokové schéma algoritmu. Převzato z [18].

Vzor zásadně ovlivňuje výsledky systému. Pro vytvoření vzoru je použita lineární kombinace eigenfaces viz kapitola 3.4.1. Takto vytvořený vzor má minimální úroveň detailu a tak je maximálně obecný viz obrázek 3.9. Dodatečně pro snížení chybné detekce byl přidán vzor oblasti očí. Ten byl vytvořen obdobným způsobem jako vzor tváře.



Obr. 3.9 Vzor pro oči vlevo a vzor pro obličej vpravo. Převzato z [18].

Prvním krokem je segmentace kůže ve vstupním barevném obrazu. Druhým krokem je křížová korelace vstupního, černobílého obrazu se vzory. Výstupem je množina odezvy na vzory nacházející se na jednotlivých pozicích a pixely obsahující barvu kůže.

Tyto příznaky jsou zpracovány v bloku False/Repeated Hit Removal. Zde jsou vyhodnoceny odezvy na vzory na základě adaptabilního prahu pro každý vstupní obraz, který se je schopen se přizpůsobit různým světelným podmínkám i šumu v obraze. Práh je stanoven ze statistických hodnot vypočtených z obrazu. Je posouzeno, zda místa s vysokou odezvou na vzory jsou na stejné pozici v obraze jako oblasti pixelů s barvou kůže. Takto jsou eliminovány false positives.

Pro odhalení vícenásobně detekovaných obličejů slouží tzv. Clustering blok, který vyhodnocuje váhu jednotlivých detekcí z bloku False/Repeated Hit Removal a případně sloučí násobně detekované obličeje tím, že ponechá jen obličeje v nevyšších vahách.

Algoritmus se takto opakuje třikrát, pro různé velikosti vstupního obrazu, což je zajištěno blokem Multi-resolution processing.

Autoři algoritmu dosáhli detekčního poměru 89%, časovou náročnost algoritmu ani výpočetní nároky neuvádí. Vlastností tohoto algoritmu je poměrně přesné určení polohy tváře. Za nevýhodu je možné považovat nutnost barevného i černobílého vstupního obrazu.

### 3.4 Metody založené na vzhledu

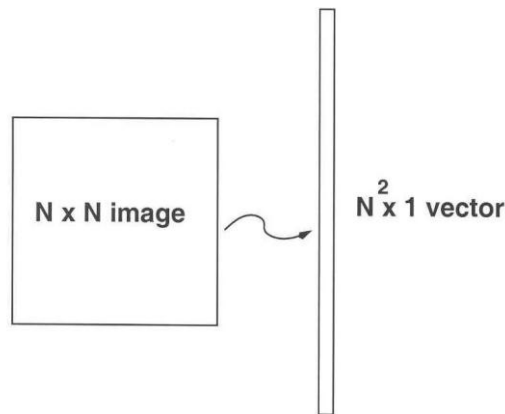
Metody založené na vzhledu jsou charakteristické tím, že modely obličejů jsou vytvořeny trénováním algoritmu [30]. Modely obličejů jsou obvykle ve formě pravděpodobnostních modelů nebo diskriminantních funkcí [14]. Při detekci jsou části vstupního obrazu srovnány s modelem obličeje.

#### 3.4.1 Eigenfaces

Turk a Pentland ve [12] uvedli metodu Eigenfaces v kombinaci s metodou vyhledání barvy kůže. Stejný přístup je uveden i v [11].

Významnou roli ve všech metodách založených na vzhledu sehrává trénovací množina obrazů. U metody Eigenfaces prezentované v [12] a [11] trénovací množina obsahuje vybrané obličejů. Zpracováním obrazů obličejů z trénovací množiny vzniknou eigenfaces.

Pro trénovací množinu je potřeba vytvořit obrazy se stejným rozměrem, s normovaným jasnem a tváře musí být v obrazech vycentrované.



Obr. 3.10 Vytvoření vektoru  $\Gamma$ . Převzato z [12].

Obraz, který je digitálně reprezentovaný maticí, je převeden na sloupcový vektor  $\Gamma$  viz obrázek 3.10.

Vektor má délku

$$\Gamma = N \cdot N, \quad (3.4)$$

kde  $N$  je rozměr obrazu.

Ze všech obrazů v trénovací množině je vypočten průměrný vektor  $\psi$  dle:

$$\psi = \frac{1}{M} \sum_{i=1}^M \Gamma_i, \quad (3.5)$$

kde  $M$  je počet obrazů v databázi a  $\Gamma_i$  jsou vektory jednotlivých obrazů z trénovací množiny.

Průměrný vektor  $\psi$  je odečten od každého obrazu  $\Gamma_i$  v trénovací množině dle:

$$\Phi_i = \Gamma_i - \psi, \quad (3.6)$$

Kde  $\psi$  je průměrný vektor,  $\Gamma_i$  jsou vektory jednotlivých obrazů z databáze a  $\Phi_i$  jsou obrazy obličejů s rysy charakteristickými pouze pro konkrétní tvář.

Od každého obličeje byly odečteny ty rysy a obličejové prvky, které jsou pro většinu obličejů společné. Vznikne tak nová databáze obrazů s obličejí s rysy specifickými pro danou tvář [11], což je princip, systémů pro rozpoznávání tváří, které používají metodu eigenfaces.

Obrazy v nově vzniklé databázi jsou reprezentovány vektory. Z těchto vektorů jsou vybrány pouze ty vektory, které odpovídají obličejům s nejvýraznějšími rysy. Tyto vektory jsou nazývány eigenvektory. Je zřejmé, že při správné volbě obrazů pro trénovací množinu, je možné lineární kombinací eigenvektorů vytvořit jakoukoliv tvář<sup>3</sup>. Takto je vytvořena sada „vlastních tváří“ neboli eigenfaces.



Obr. 3.11 Množina obrazů  $\Phi$  od kterých je odečten průměrný vektor. Převzato z <http://www.cs.princeton.edu/~cdecero/eigenfaces/>.

Při detekci se algoritmus snaží vytvořit obdobný obraz, který se nachází v oblasti, která byla označena jako potencionální oblast tváře. K tomu používá lineární kombinaci „vlastních tváří“ Podle míry shody ve výsledku lineární kombinace a vstupní oblasti, kde se předpokládá tvář, je rozhodnuto, zda-li se v oblasti opravdu tvář nachází.

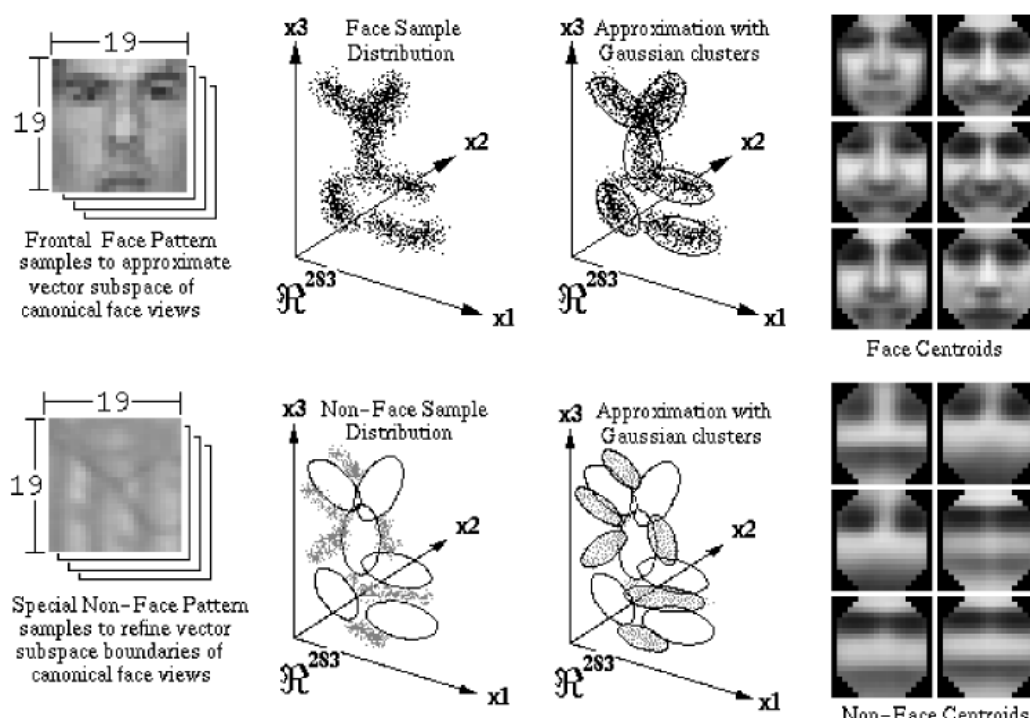
<sup>3</sup> Jedná se teoretický případ, v praxi je možné se libovolné tváří více či méně přiblížit.

### 3.4.2 Distribution-based methods

Distribution-based methods jsou metody založené na pravděpodobnostním modelu obličeje. Model slouží pro získání nejvíce relevantních příznaků pro klasifikátor.

Sung a Poggio [20] představili metodu, která během trénování vytvoří model obličeje ve vícerozměrném prostoru a při detekci porovnává vstupní obraz modelem obličeje. Výsledkem porovnání modelu obličeje a vstupního obrazu je získání příznaků. Všechny pixely vstupního obrazu se projeví při reprezentaci obrazu v prostoru. Každý pixel vstupního obrazu má vliv na příznaky. Klasifikaci tedy ovlivňují všechny pixely.

Model obličeje je získán trénováním algoritmu. Trénovací množina obsahuje obrazy o rozměru 19x19 pixelů (ve [20] je tento rozměr zvolen jako základní buňka).



Obr. 3.12 V horní řadě postupně zleva, množina obrazů tváře v trénovací množině, reprezentace tváří v mnohadimenzionálním prostoru, clusterované shluky reprezentací obrazů tváří a nejvíce pravděpodobné tváře, tj. ty, které jsou v těžistích. V druhé řadě postupně zleva, množina obrazů okolí v trénovací množině, jejich reprezentace v mnohadimenzionálním prostoru společně s clusterů tváří, clusterované reprezentace tváří i okolí, těžistě clusterů obrazů okolí. Převzato z [20].

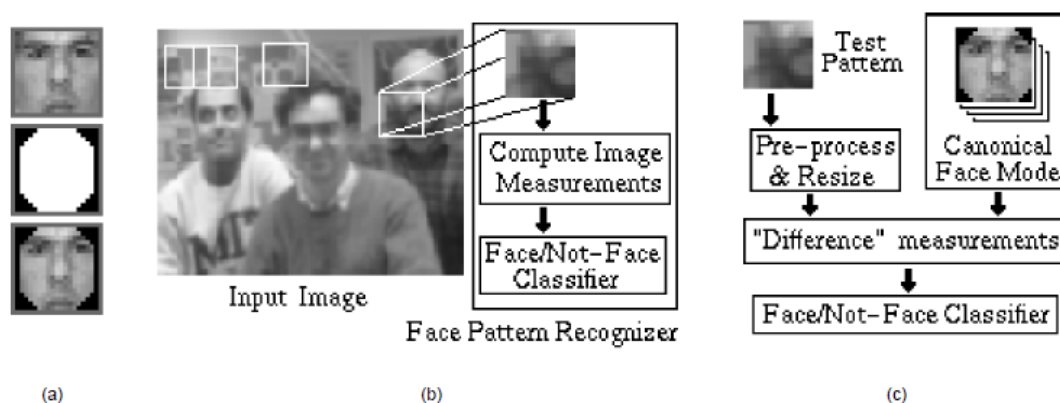
Trénovací množina zahrnuje obrazy obličeje a obrazy „podobné“ obličeji (viz. obrázek 3.12, Non-face centroids). Algoritmus postupně vytvoří model obličeje v prostoru (viz. obrázek 3.12, Face Sample Distribution). Jednotlivé obrazy obsahující obličej jsou prostoru reprezentovány body.

Z obr. 3.12 je vidět, že body vytvoří v prostoru určité shluky. Shluky jsou clusterovány do celkem dvanácti clusterů (viz. obrázek 3.12, Approximation with Gaussian Clusters), šest clusterů tvoří body reprezentující obrazy obličejů a šest clusterů tvoří body reprezentující obrazy okolí. Každý cluster tvarově odpovídá vícerozměrnému

Gaussovu rozdělení, které není ve všech osách symetrické. U clusterů je určen bod, ve kterém je nejvyšší výskyt – těžiště clusteru a je vypočtena kovarianční matice, popisující okolí těžiště clusteru.

Prvním krokem detekce obličeje je předzpracování subobrazů vytvořených ze vstupního obrazu. Provede se korekce gradientu jasu pro zmírnění vlivu výrazných stínů v obličeji, dále je provedena ekvalizace histogramu. Je použita maska, která vybere jen relevantní pixely subobrazu viz obrázek 3.13 a). Maska sníží i počet zpracovávaných pixelů z 361(19x19) na 283. Výše uvedené operace je nutné provést se všemi vstupními subobrazy.

Proces zikávání příznaků je proveden převedením vstupního předzpracovaného subobrazu rozměru na vektor, tento vektor s počtem vzorků 283 (uvažujeme jen relevantní pixely) je reprezentován bodem v 283 dimenzionálním vektorovém prostoru. Dále následuje proces měření vzdáleností mezi bodem reprezentující vstupní obraz a těžišti dvanácti clusterů. Tyto vzdálenosti jsou příznaky pro klasifikátor. Pro měření je využito normalizované Mahalanobisovy vzdálenosti.



Obr. 3.13 Obrázek a) znázorňuje postupně vytvoření trénovací množiny obrazů tváře, postupně, výběr obličeje, maska, použitá maska, pro snížení dimenzionality prostoru a vyjmutí irelevantního okolí tváře. b) zobrazuje vstupní obraz rozdělený na překrývající se subobrazy. c) vývojový diagram algoritmu, který prezentovali Sung a Poggio[20]. Převzato z [20].

Jako klasifikátor je použit MLP- multi-layer perceptron-vícenásobný perceptron. Jedná o síť buněk, která na základě vstupní informace klasifikuje danou reprezentaci vstupního obrazu. Síť má jednu výstupní buňku, která nabývá hodnotu 1, pro obraz klasifikovaný jako tvář a 0 pro ostatní.

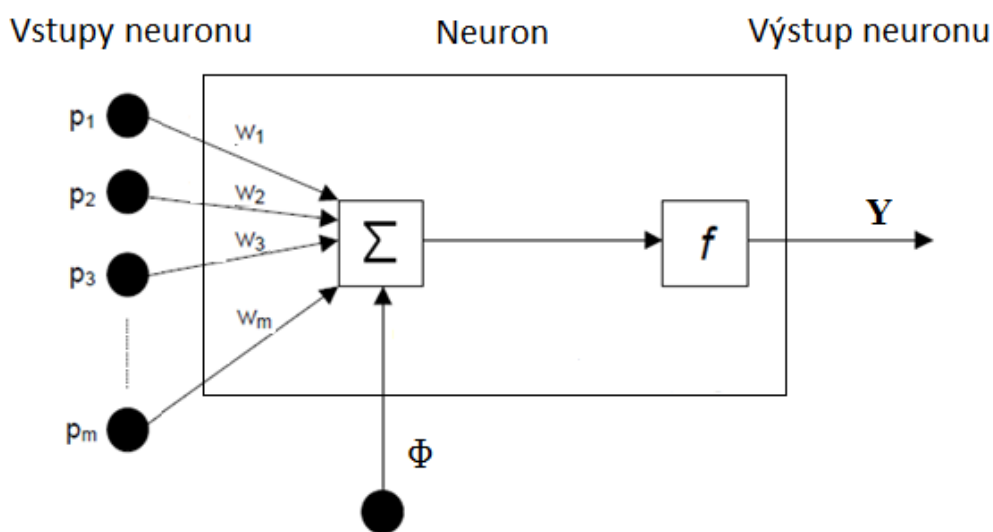
Metoda byla testována na obrazech s vysokým rozlišením a výraznou variancí jasu, ale i na obrazech s nízkou kvalitou. Při testování na databázi s kvalitními obrazy byl detekční poměr 96,3% , u databáze s nízkou kvalitou byl detekční poměr 79,9%. V druhém případě byla nízká detekce způsobena extrémními jasovými podmínkami v obrazu nebo velmi špatnou kvalitou. Na databázi 301 kvalitních obrazů byly pouze 3 false positives. Rychlost algoritmu není uvedena.

Trénovací množina obrazů tváře obsahovala 4150 obrazů, kde 1067 je reálných a zbytek je uměle vytvořených. Umělé obrazy jsou zrcadlově převrácené nebo mírně natočené. Všechny obrazy byly uměle vyřezány

### 3.4.3 Neural networks –Neuronové sítě

Tato skupina metod využívá jako klasifikátor neuronové sítě. Umělé neuronové sítě jsou počítačovou analogií lidského mozku. Umělé neuronové sítě mají podobnou strukturu a buňky jako lidský mozek.

Neuronová síť je síť vzájemně propojených neuronových procesorů [23], které jsou paralelně propojeny. Neurony si vzájemně předávají informace a transformují je [31]. Neuron je charakterizován přenosovou funkcí. Neuron může mít libovolné množství vstupů, ale má jen jeden výstup. Schéma neuronu je na obr. 3.14.



Obr. 3.14 Blokové schéma jednoho neuronu. Převzato z [23] a modifikováno.

Na obrázku 3.14 jsou  $p_i$  vstupy neuronu, které jsou váhovány koeficienty  $w_i$ . V bloku  $\Sigma$  je proveden součet váhovaných vstupů  $p_i$ . Pro aktivaci neuronu je potřeba, aby součet váhovaných vstupů nabýval větší hodnoty než práh  $\Phi$ . Tuto skutečnost vyjadřuje rovnice [31]:

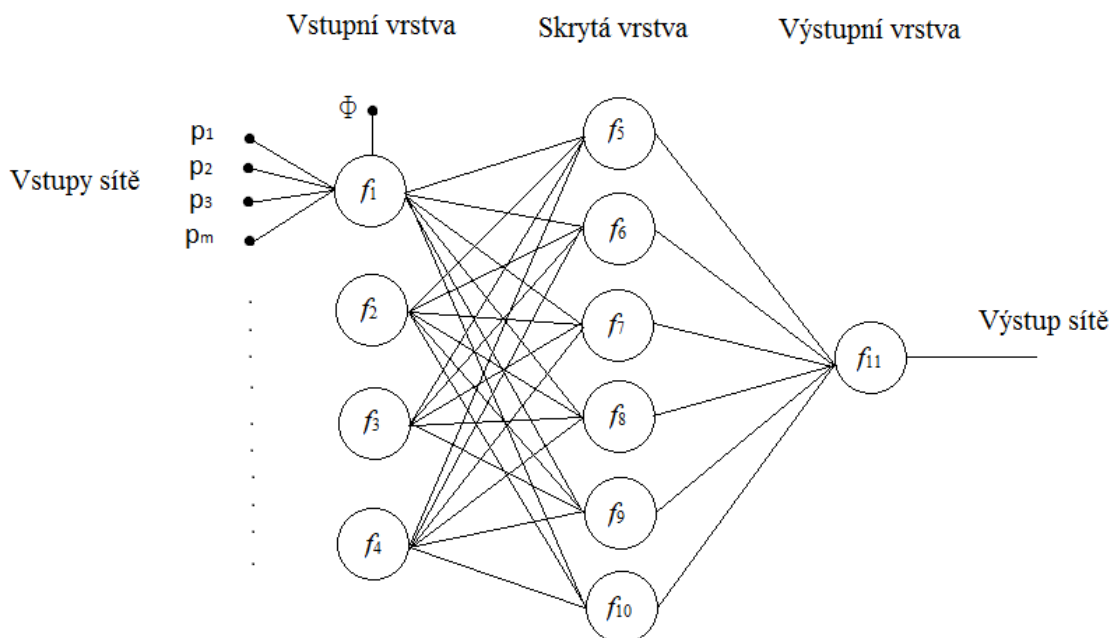
$$\sum_{i=1}^m (p_i w_i) > \Phi. \quad (3.7)$$

Je-li neuron aktivován, je jeho výstupní hodnota  $Y$  definována rovnicí [31]:

$$Y = f\left(\sum_{i=1}^m (p_i w_i) + \Phi\right), \quad (3.8)$$

kde  $p_i$  jsou vstupy neuronu,  $w_i$  jsou váhové koeficienty,  $\Phi$  je aktivační práh,  $f$  je typ přenosové funkce a  $m$  je počet vstupů. Jeden neuron je možné použít pro řešení jednoduchých problémů.

Více komplexní problémy jsou řešeny soustavou neuronů, tj. neuronovými sítěmi. Základní struktura neuronové sítě je na obrázku 3.15. Neurony, které nepředstavují vstupy sítě (nejsou ve vstupní vrstvě) nebo nejsou výstupními buňkami celé sítě (nejsou ve výstupní vrstvě), jsou v takzvaných skrytých vrstvách [23]. Skrytých vrstev může být i více.



Obr. 3.15 Schéma neuronové sítě.

Pro detekci obličeje je možné použít mnoho různých konstrukcí neuronových sítí. Nevhodná konstrukce sítě má za následek špatné výsledky celého algoritmu. Konstrukce je tedy během vývoje algoritmu optimalizována (počet vrstev, počet uzlů) [13].

Neuronovou síť pro potřeby detekce obličeje je nutné natrénovat. Trénování v tomto případě znamená nastavení hodnot váhových koeficientů  $w_i$ , viz. obrázek 3.14 a 3.15. Trénování probíhá, dokud není dosaženo požadovaného výsledku. Po dokončení trénování je síť schopna klasifikovat zcela nová neznámá vstupní data.

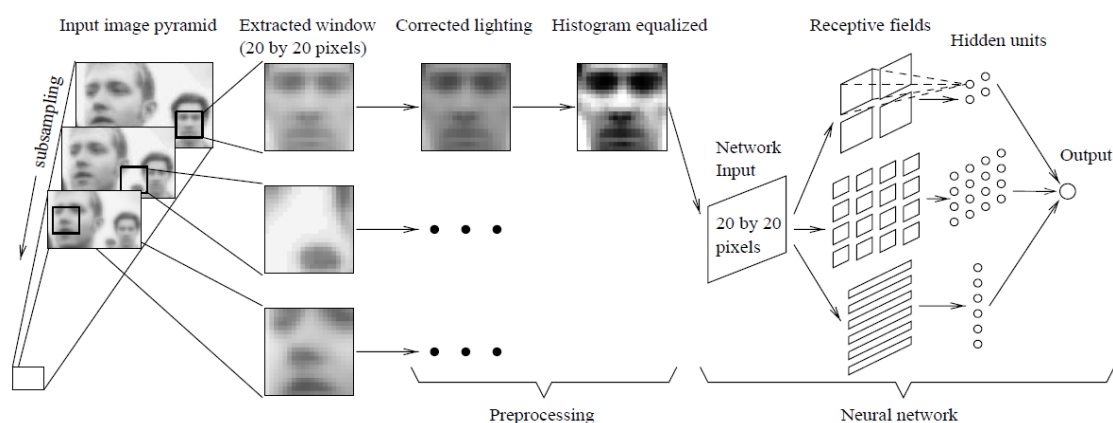
Rowley, Baluja a Kanade [22] využili výše popsaného klasifikátoru pro detekci obličeje. Na vstupní obraz použijí několik filtrů<sup>4</sup> – neuronových sítí. Výstup jednotlivých sítí je vyhodnocen v druhé fázi algoritmu. V [22] je uvedeno, že pro odstranění false positives je možné použít i více neuronových sítí stejné konstrukce, ovšem jinak trénovaných. V [22] je uvedeno několik modifikací klasifikátoru. Zde bude uveden algoritmus využívající dvě neuronové sítě.

Trénovací množina neuronových sítí v [22] sestává z 1050 vybraných z různých databází tváří a z WWW. Všechny obličeje v trénovací množině byly ručně označeny.

Před vlastní filtrací neuronovou sítí jsou vstupní subobrazy (části vstupního obrazu

<sup>4</sup> Ve [20] se pojednává o neuronových sítích jako o filtrech (neural network-based filters)

s rozměrem 20x20 pixelů, viz. obrázek 3.16 vlevo) předzpracovány. (viz. obrázek 3.16, preprocessing). Jsou odstraněny stíny a přechody jasu v rámci vybraného subobrazu. Ve vybraném subobrazu je vyhodnoceno rozložení jasu a subobrazu je přiřazeno korekční rozložení jasu ( viz obrázek 3.17, best fit linear function), vybrané korekční rozložení jasu je následně odečteno od intenzit jasu subobrazu, (výsledek odstranění jasových přechodů viz. obrázek 3.17, lighting corrected window). Dále je použita maska pro vyjmutí irelevantního okolí tváře (viz. obrázek 3.17, oval mask for ignoring background pixels). Poslední fází předzpracování je vyrovnání histogramu subobrazu (viz. obrázek 3.17, histogram equalized window). Předzpracované subobrazy vstupního obrazu vstupují do neuronových sítí (viz. obrázek 3.16 neural network).



Obr. 3.16 Postup algoritmu, který prezentovali Rowley, Baluja a Kanade. Převzato z [22].

Neuronová síť navržená v [22], má 3 různé, nezávislé, skryté vrstvy neuronů (viz. obrázek 3.16, hidden units). Vstupní subobraz je dělen třemi poli receptorů (viz. obrázek 3.16, receptive fields). Receptorová pole poskytnou informace jednotlivým neuronům. Každé skryté vrstvě neuronů jsou přiděleny informace z odpovídajícího pole receptorů. Horní skrytá vrstva obsahuje čtyři neurony. Tato vrstva má detekovat samostatné oko, nos a kraje úst. Dolní skrytá vrstva obsahuje pět neuronů. Tyto neurony detekují pár očí a ústa. Střední vrstva obsahuje 16 neuronů, ty detekují jemné obličejové rysy. Celá neuronová síť má jeden výstupní neuron, který klasifikuje zkoumaný subobraz.

Tento postup se opakuje pro všechny subobrazy. Pokud je použito více neuronových sítí, každý subobraz je zpracován všemi sítěmi.



**Oval mask for ignoring  
background pixels:**



**Original window:**



**Best fit linear function:**



**Lighting corrected window:  
(linear function subtracted)**



**Histogram equalized window:**



Obr. 3.17 Předzpracování vstupního obrazu. Převzato z [22].

Pro detekci tváří různého rozměru je vstupní obraz podvzorkován a celý algoritmus se iterativně opakuje.

Tímto iterativním postupem získáme množinu detekovaných obličejů v různých měřítkách a na různých pozicích obrazu.. V této množině se vyskytují false positives a násobně detekované obličeje od jedné nebo více neuronových sítí. Z této množiny je potřeba vyloučit false positives a výsledky neuronové sítě zpracovat.

V druhé fázi algoritmu jsou odstraněny false positives a označeny obličeje. Pro prohlášení oblasti jako tváře je nutná násobná detekce oblasti. Je specifikován počet násobných detekcí na jedné pozici, který musí být dosažen, potom je oblast prohlášena za obličej. Takto je vymezena oblast obličeje. Všechny ostatní detekované obličeje, které do této oblasti zasahují, jsou prohlášeny za chyby a jsou odstraněny. Tímto způsobem jsou chybně vyloučeny i tváře, které mohou být částečně zakryté výraznější tváří.

Posledním krokem druhé fáze algoritmu je odstranění false positives. Každá neuronová síť je trénována na jiné trénovací množině [23] (pokud je použito více sítí). Zřejmě i false positives jednotlivých sítí budou na jiných pozicích a v jiných měřítkách. Takto jsou false positives odstraněny.

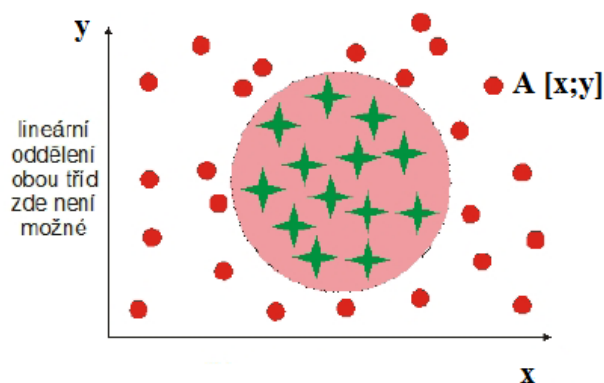
Neuronové sítě ve [23] detekují hlavně oči, nos a ústa. V citlivostní analýze metody je uvedeno, že pro detekci stačí pouze oči. Pro testování metody byly použity dvě databáze obrazů. První byla databáze byla sestavena z obrazů z CMU (webová databáze testovacích obrazů), dále obsahovala obrazy z WWW, naskenované fotografie a novinové obrázky. Velmi rozmanitá kvalita obrazů a komplexní pozadí jsou vhodné pro testování false positives rate. Druhá testovací databáze obsahovala vybrané obrazy z databáze FERET. Databáze obsahuje obrazy s obličejí v různých úhlech, tyto byly použity pro testování citlivosti detektoru na úhel natočení obličeje. Rowley, Baluja a Kanade testovali algoritmus v mnoha modifikacích a všechny výsledky detailně analyzovali v [23].

Detekční poměr se pohybuje dle modifikace od 77,9% do 90,3% s nízkým počtem false positives. Počet false positives je úměrný detekci. Algoritmus byl testován z hlediska rychlosti, po prvních testech byl modifikován a výsledná rychlost byla 2-4 sekundy, za použití dvou neuronových sítí. Vstupní obraz měl rozměr 320x240 pixelů.

### 3.4.4 Support vector machine

Support vector machines (SVM) je algoritmus, který rozdělí vstupní data do dvou skupin vytvořením lineárního klasifikátoru (rovinu), který je umístěn mezi skupinami v mnohodimenzionálním prostoru. Základní myšlenkou této metody je, že jakékoliv skupiny bodů, obecně v prostoru, je možné rozdělit lineárním klasifikátorem (rovinou) ve více dimenzionálním prostoru. Takovýmto způsobem je možné reprezentovat velice složité nelineární funkce [26].

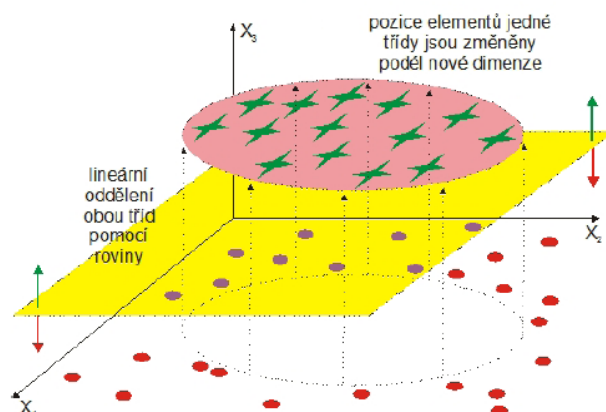
Trénování klasifikátoru SVM znamená sestavení lineárního klasifikátoru v prostoru mezi dvěma skupinami bodů. Při trénování klasifikátoru je zadána množina vstupních dat. Ta je zobrazena v prostoru s počtem dimenzí, který odpovídá počtu souřadnic vstupních bodů (např. bod A o souřadnicích  $x;y$  je zobrazen v rovině viz. obrázek 3.18).



Obr. 3.18 Zobrazení dvou množin bodů v rovině. Převzato z [26] a upraveno.

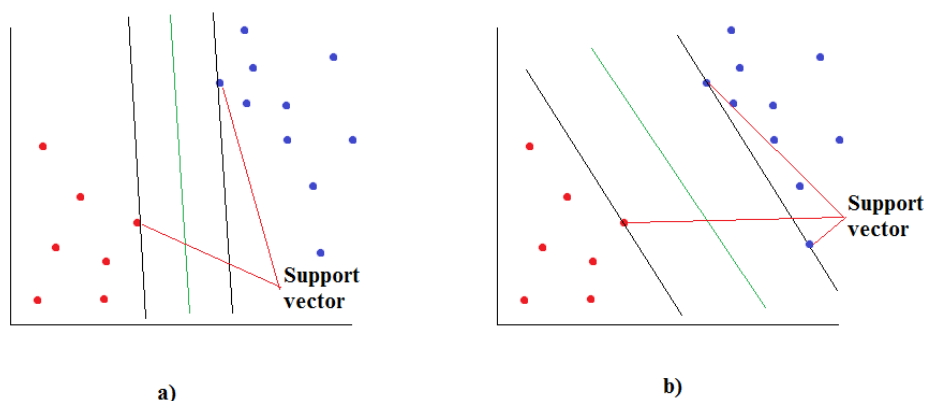
Jak je vidět na obrázku 3.18, není možné oddělit červené tečky od hvězdiček pomocí lineárního klasifikátoru (přímky). Pokud ovšem bude stanovena vhodná reprezentace červených teček a hvězdiček ve více dimenzionálním prostoru, je možné je oddělit rovinou, takovýto příklad zjednodušeně dokladuje obrázek 3.19.

Z obrázku 3.20 a 3.19 je zřejmé, že klasifikátor reprezentovaný v rovině kružnicí, je možné v prostoru nahradit rovinou.



Obr. 3.19 Zobrazení dvou množin dat ve vícedimenzionálním prostoru a nalezení oddělující roviny. Převzato z [26] a upraveno.

Při trénování je nalezen takový prostor, ve kterém jsou dvě množiny separovatelné lineárním klasifikátorem. Nicméně je třeba vzít v úvahu, že v různých prostorech je možné zkonstruovat jiné klasifikátory s jiným vztahem k reprezentovaným bodům. Cílem se nalezení optimálního prostoru pro klasifikaci. Za optimální prostor, můžeme považovat takový prostor kde lineární klasifikátor, nejspolehlivěji oddělí skupiny. To geometricky znamená, že klasifikátor má největší vzdálenost od obou skupin bodů [24].



Obr. 3.20 Možné oddělení dvou skupin bodů pomocí různých lineárních klasifikátorů. Převzato z [24] a modifikováno.

Z obrázku 3.20 je zřejmé, že klasifikátor zkonstruovaný v situaci b), skupiny oddělí s větší vzdálenostní rezervou.

Klasifikátor (na obrázku 3.20 zobrazen zelenou přímkou) je umístěn mezi nejbližšími body skupin. Tyto body, které jsou nejbližší klasifikátoru, tvoří hranici skupiny (na obrázku 3.20 je hranice skupiny zobrazena černou přímkou). Body, které určují tuto hranici, jsou nazývány support vectors – pomocné vektory.

Rozšíření prostoru je obvykle realizováno pomocí kernel functions – jádrových funkcí. Tyto funkce jsou schopné vytvořit reprezentaci bodů ve více dimenzionálním prostoru.

SVM byl použit pro detekci obličeje v práci, kterou prezentovali Osuna, Freud a Girosi [24]. Postup této metody je obdobný jako v předchozích případech. Vstupní obraz je rozdělen na subobrazy 19x19 pixelů, proběhne předzpracování obrazu. Předzpracovaný obraz je klasifikován SVM.

Metoda navržená v [24] je testována dvojicí databází. První databáze 313 kvalitních obrazů s jedním obličejem v každém obraze, druhá databáze 23 obrazů různé kvality s celkem 155 obličejů. Ve [24] jsou srovnány výsledky metody používající SVM [24] a metody, kterou prezentovali Sung a Poggio [20]. Metoda využívající SVM má na první databázi detekci 97,1% a na druhé 74,2%. Dosahuje tak celkově většího detekčního poměru než metoda uvedená ve [20]. Počet chyb false positives u první databáze byl čtyři, u druhé dvacet. V obou případech je to více než dosahovala metoda, kterou prezentovali Sung a Poggio [20]. Nicméně algoritmus využívající SVM je přibližně třicetkrát rychlejší než algoritmus prezentovaný ve [20]. Počet pomocných vektorů se pohybuje okolo 10 000.

### **3.4.5 Další appearance-based methods**

Skupina appearance-based methods je široká a není možné zde popsat všechny metody. Zde jsou uvedeny další klasifikátory, které je možné použít pro detekci obličeje:

Viola-Jones detektor

SNoW (Sparse Network of Windows).

HMM (Hidden Markov Model).

Naive Bayes Classifier.

## **3.5 Výsledky metod**

Jak již bylo uvedeno v kapitole 2.2 a ve [14], není možné vyhodnotit metodu jako nejlepší nebo „nejúspěšnější“. Výsledky metod jsou zásadně ovlivněny trénovací množinou a testovací množinou obrazů. Pro různé testovací množiny může jedna metoda vykazovat rozdílnou detekci až o desítky procent. Obvykle se testy provádí na rozšířených databázích FERET, MIT a CMU. V tabulce 2 jsou uvedeny dosažené výsledky metod.

Metoda	Autoři	Detekce (%)	False positives	Pozn.
Rule-based face detection	Koutropolis,Pitas [15]	86,5	-	Nejpřísnější testování
Probabilistic feature grouping	Yow, Cipolla [17]	92%	-	-
Skin color	Chen, Lin [9]	92%		Na 1,4GHz PC, 10fps <sup>5</sup>
Násobné rysy	Marius, Pennathur, Rose [10]	95%	4,3%	Čas 100s
Predefined templates	Ping,Weng,Lau [18]	89%	-	-
Distribution based method	Sung, Poggio [20]	96,3%	-	
Neural network	Rowley, Baluja, Kanade	77,9-90,3%		čas výpočtu 2-4s
SVM	Osuna,Freund,Girosi [24]	97,1% , 74,2% <sup>6</sup>	4/313, 20/23 <sup>7</sup>	
Viola-Jones	Viola, Jones [1]	93%		

Tabulka 2 Přehled výsledků metod pro detekci obličejů.

Za obecnou nevýhodu, při použití učících se klasifikátorů jako jsou neuronové sítě, SMV a adaboost, můžeme považovat vytváření databází s velkým počtem obrazů, řádově tisíce obrazů.

Na robustnost metod má vliv volba příznaků a způsob předzpracování vstupních obrazů. Metody založené na invariantních rysech při nízké kvalitě obrazu spoléhají na detekci alespoň některých rysů. Metody založené na vzhledu uvažují a hodnotí celý vstupní obraz nebo jeho části. Z hlediska robustnosti, je princip metod založených na vzhledu, schopen zajistit lepší výsledky.

<sup>5</sup> Fps – frames per second, fps udává počet zpracovaných obrazů za sekundu

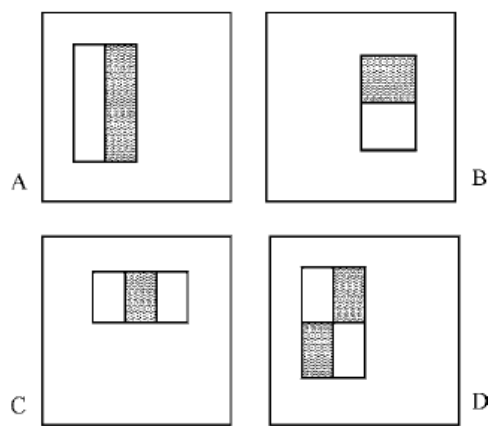
<sup>6</sup> Testováno na dvou databázích

<sup>7</sup> 4/313 Vyjadřuje, čtyři false positives na 313 vstupních obrazech.

## 4 DETEKTOR VIOLA-JONES A JEHO IMPLEMENTACE

### 4.1 Příznaky – Haarovy filtry

Jako příznaky jsou použity odezvy na Haarovy filtry (Haar-like features). Haarovy filtry jsou plošnými detektory charakteristických obličejových rysů. Dva filtry jsou hranové detektory, viz Obr 4.1 A a B. Jeden filtr je použit pro detekci čar, viz Obr 4.1 C a jeden je použit pro detekci složitějších rysů viz Obr 4.1 D.



Obr.4.1 Haarovy filtry. A horizontální (hranový) filtr, B vertikální (hranový) filtr, C trojnásobný (čárový) filtr, D čtyřnásobný (diagonální) filtr. Převzato z [1].

Haarových filtrů může být mnoho druhů, složitější filtry mohou pomoci detekovat obličeje v obecnějších polohách a podmínkách. V této práci jsou použity pouze filtry zobrazené na obrázku 4.1.

Odezva na Haarův filtr – příznak, je číslo, které je výsledkem rozdílu sum jasových hodnot pod světlou a tmavou částí Haarových filtrů, viz Obrázek 4.2.

V obraze je vypočteno mnoho takovýchto odezev-příznaků, na jejichž základě je rozhodnuto o přítomnosti tváře.

## 4.2 Integrální obraz

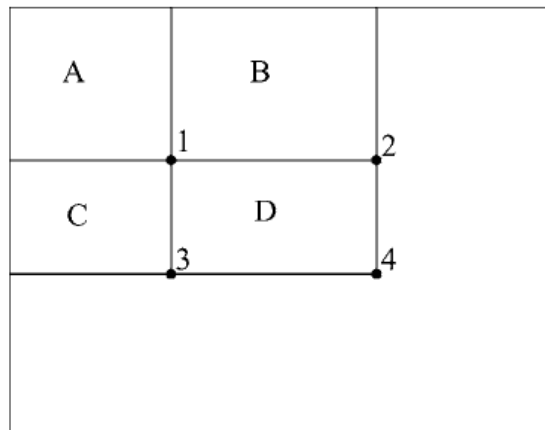
Počítat sumu jasových hodnot na každé pozici, pro každý filtr by bylo výpočetně náročné. Proto je použito tzv. Integrálního obrazu (integral image). Integrální obraz je reprezentace vstupního obrazu, v níž jsou jasové hodnoty jednotlivých pixelů kumulativně sečteny v řádcích a sloupcích. Tato skutečnost je vyjádřena rovnicemi:

$$s(x, y) = s(x, y-1) + i(x, y), \quad (4.1)$$

$$ii(x, y) = ii(x-1, y) + s(x, y), \quad (4.2)$$

kde  $s(x,y)$  je kumulovaný součet jasových hodnot v řádku a  $i(x,y)$  jsou hodnoty původního vstupního obrazu a  $ii(x,y)$  je integrální obraz. V integrálním obrazu je tedy na pozici  $x,y$  součet všech jasových hodnot původního obrazu od pozice  $x,y$  k levému hornímu rohu obrazu.

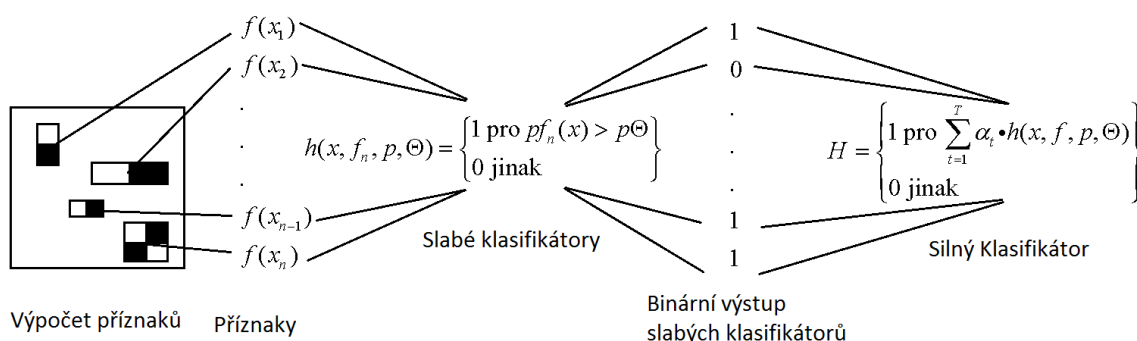
Chceme-li například vypočítat součet hodnot v obdélníku D, stačí použít pouze hodnoty integrálního obrazu ležící v rozích obdélníka D. Hodnota integrálního obrazu na pozici 1 je suma hodnot v obdélníku A, 2 je suma hodnot z A+B, 3 je suma hodnot A+C. D bude vypočteno jako  $4-2-3+1$ .



Obr. 4.2 Výpočet jasových hodnot v obdélníku D. Převzato z [1].

### 4.3 Klasifikátor

Princip detekce spočívá výpočtu informativních příznaků. Necht' je informativní příznak jakýkoliv příznak, pomocí něhož je možné rozlišit tvář od pozadí s úspěšností větší než 50%. Klasifikátor, který využívá informativní příznaky, nazveme slabý klasifikátor, jeho klasifikační schopnost je větší než 50%. Pokud použijeme více slabých klasifikátorů, pak je možné ve výsledku dosáhnout vysoké klasifikační schopnosti. Slabé klasifikátory jsou spojeny v silném klasifikátoru. Vstupem silného klasifikátoru je váhovaný výstup slabých klasifikátorů. Tato skutečnost je vyobrazena na obrázku 4.3.



Obr. 4.3 Schéma klasifikace

Detektor Viola - Jones využívá kaskádu silných klasifikátorů. Každý klasifikátor z celé kaskády má jiné klasifikační schopnosti, protože používá jiné slabé klasifikátory s jinými příznaky. Kaskáda slouží pro urychlení procesu klasifikace. První klasifikátor v kaskádě odmítne ty subobrazy, kde není obličej, další klasifikátor již ohledává ty subobrazy, které nebyly dosud vyloučeny jako pozadí. Opět vyloučí další subobrazy, které neodpovídají tváři. Po průchodu kaskádou klasifikátorů zbude jen subobraz, kde se nachází tvář.

Příznaky pro klasifikátor jsou tedy vypočteny z integrálního obrazu. Příznak  $f(x)$  vstupuje do slabého klasifikátoru:

$$h(x, f, p, \Theta) = 1 \text{ pro } pf(x) < p\Theta, \text{ jinak } 0, \quad (4.3)$$

kde  $h(x, f, p, \Theta)$  je binární výstup slabého klasifikátoru,  $f(x)$  je příznakem,  $\Theta$  je práh a  $p$  značí směr nerovnosti.



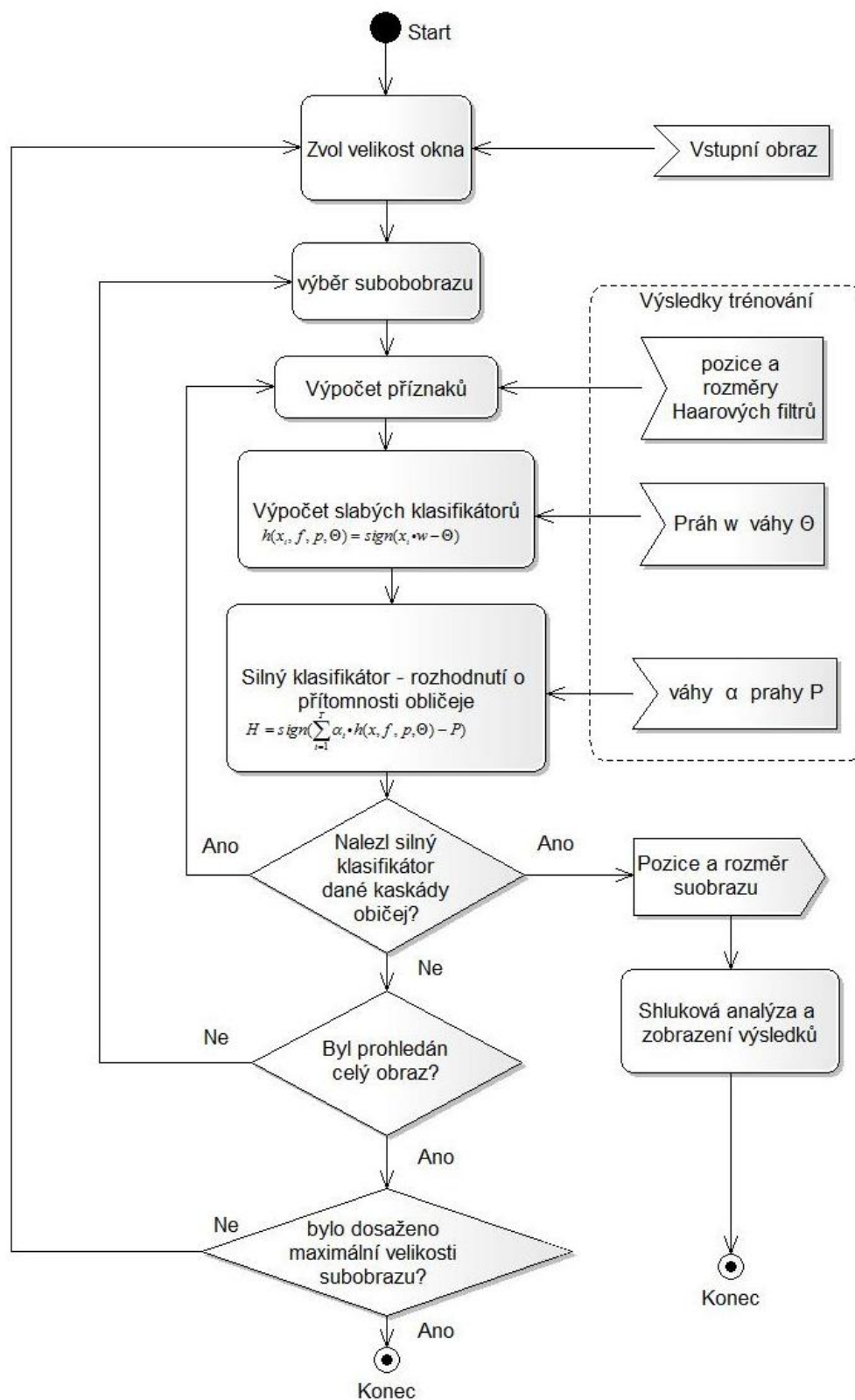
Slabé klasifikátory vstupují do silného klasifikátoru-perceptronu, jejich váhy  $\alpha_t$  jsou úměrné chybovosti jednotlivých slabých klasifikátorů při trénování. Můžeme tedy říci, že každý slabý klasifikátor má na celkovou klasifikaci vliv úměrný jeho chybovosti. Silný klasifikátor potom má tvar:

$$H = \text{sign}\left(\sum_{t=1}^T \alpha_t \cdot h(x, f, p, \Theta) - P\right), \quad (4.4)$$

kde  $\alpha_t$  jsou váhy slabých klasifikátorů,  $P$  je práh daného silného klasifikátoru vypočteny při trénování,  $T$  je celkový počet slabých klasifikátorů, ze kterých je složen silný klasifikátor.

Celá kaskáda se skládá z několika takovýchto silných „monolitických“ klasifikátorů výše popsaným způsobem.

## 4.4 Vývojový diagram detektoru



Obr. 4.4 Obecný vývojový digram detektoru Viola-Jones.

## 4.5 Trénování algoritmu - Adaboost

Adaboost je tzv. trénovací algoritmus. Trénovací algoritmy obecně mají za úkol na základě statistického vyhodnocení nalézt informativní příznaky nebo hodnoty nutné ke klasifikaci. V případě detektoru Viola-Jones se jedná o konkrétní informativní příznaky, prahové hodnoty slabých klasifikátorů  $\Theta$ , váhy obrazů  $w$  a váhy slabých klasifikátorů  $\alpha$ .

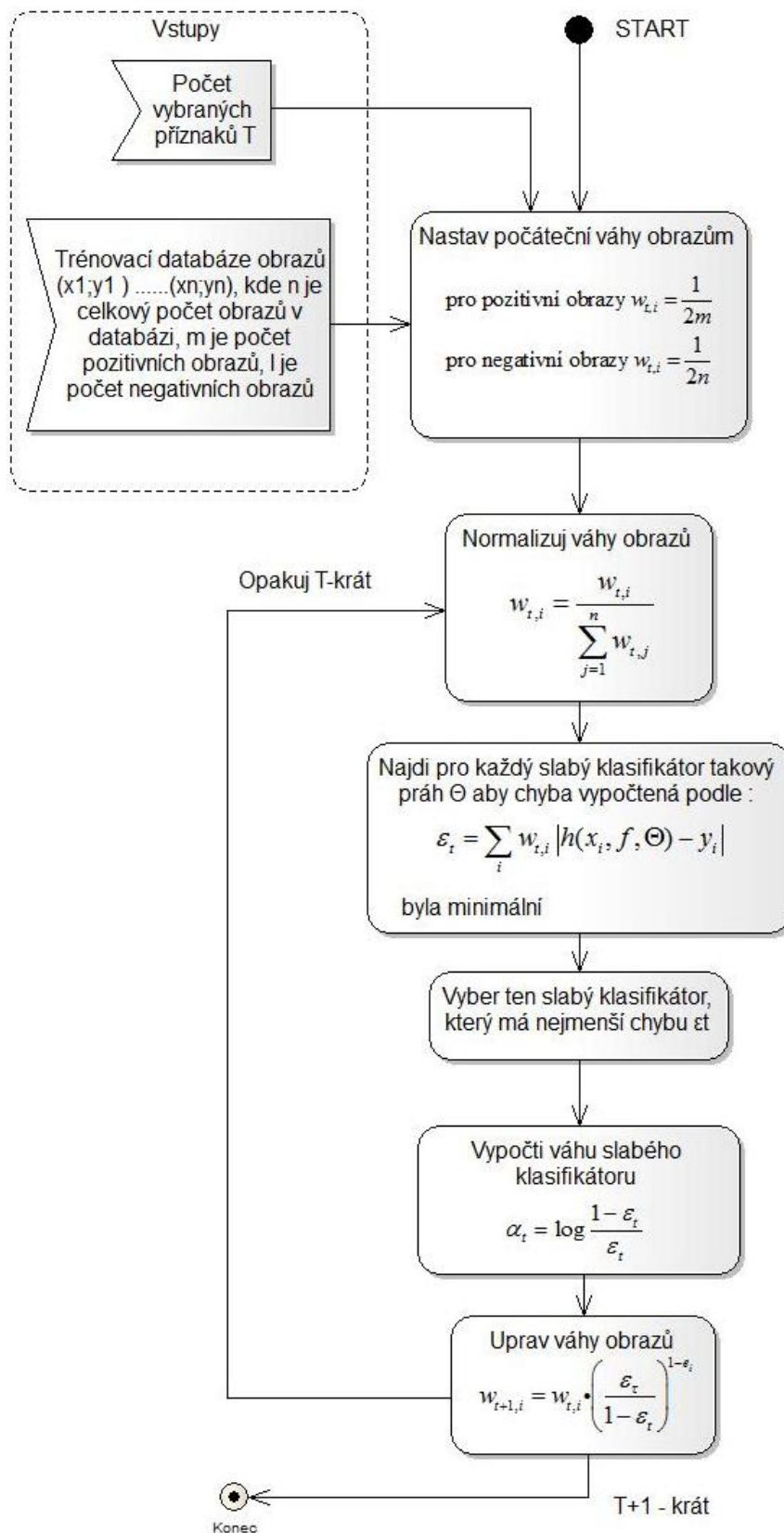
Jak bylo popsáno v kapitole 4.3, detektor Viola-Jones využívá kaskádu klasifikátorů. Kaskáda je použita pro úsporu času. Z toho ovšem plyne, že je možné sestavit i tzv. monolitický klasifikátor, který by obsahoval všechny slabé klasifikátory jako celá kaskáda, přičemž vlastnosti detektoru by zůstali stejné. Pro vytvoření monolitického klasifikátoru byl použit modifikovaný algoritmus Adaboost [1].

Pro natrénování je nutná trénovací databáze vstupních obrazů, která čítá řádově tisíce obrazů obsahujících obličeje – pozitivní obrazy a tisíce obrazů pozadí – negativní obrazy. Adaboost vybírá jednotlivé informativní příznaky, podle toho, jakou mají chybovost při klasifikaci obrazů trénovací databáze. V jednotlivých cyklech Adaboostu je provedeno: nastavení důležitosti správné klasifikace jednotlivých obrazů – váhy obrazů, vypočtení všech možných příznaků na celé trénovací množině, stanovení optimálního prahu  $\Theta$  tak, aby chybovost  $\epsilon_t$  slabého klasifikátoru používajícího daný příznak byla minimální. Výběr slabého klasifikátoru s nejmenší chybovostí a jeho zařazení mezi klasifikátory použité pro detekci. Jednotlivým obrazům jsou upravovány váhy podle toho, zda byli vybraným klasifikátorem správně klasifikované. Tento cyklus se opakuje, dokud není dosažen žádaný počet slabých klasifikátorů.

Vývojový diagram Adaboostu je zobrazen na obrázku 4.5. Podstatným parametrem celého klasifikačního procesu je počet příznaků  $T$ . Ten je v případě monolitického klasifikátoru volen na začátku trénování. Počet klasifikátorů musí být zvolen tak, aby nedošlo k přetrénování. Přetrénování je jev, ke kterému dochází při vysokém počtu klasifikátorů. Při přetrénování dojde ke ztrátě schopnosti rozlišení obecné tváře, klasifikátory jsou zaměřeny pouze na obrazy trénovací množiny [3]. Na druhou stranu je nutné zvolit takový počet klasifikátorů, aby nedocházelo k chybám typu false positives.

Významným krokem celého Adaboostu je hledání optimálního prahu  $\Theta$  pro daný klasifikátor tak, aby chybovost  $\epsilon_t$  byla minimální. Při daném prahu  $\Theta$  je možné vypočítat výstupy slabých klasifikátorů, ty nabývají hodnot  $\{0;1\}$ . Podle toho, zda na konkrétním obrazu byl nalezen obličej nebo ne viz. rovnice 4.3. Iteračním hledáním prahu  $\Theta$  je možné nalézt minimum chybovosti  $\epsilon_t$ . Tímto způsobem je možné nalézt optimální prahy pro všechny možné slabé klasifikátory a tím i jejich minimální chybovosti. Je vybrán slabý klasifikátor s celkovou minimální chybovostí.

Následně jsou upraveny váhy obrazů – míra důležitosti správné klasifikace slabým klasifikátorem vybraným v dalším cyklu. Váha obrazu se zvětší v případě že byl posledním vybraným klasifikátorem špatně klasifikován. Tuto informaci v obrázku 4.5 nese proměnná  $\epsilon_t$ , její obor hodnot je  $\{0;1\}$ , 0 pokud byl konkrétní obraz špatně klasifikován a 1 pokud byl klasifikován správně. Tímto způsobem se ve vahách obrazů projeví i již vybrané slabé klasifikátory.



Obr. 4.5 Vývojový diagram algoritmu Adaboost pro trénování monolitického klasifikátoru

Tímto způsobem je vybráno všech  $T$  slabých klasifikátorů.

### 4.5.1 Trénování kaskádního klasifikátoru

Kaskádní klasifikátor sestává z několika monolitických klasifikátorů – stupňů kaskády, s malým počtem slabých klasifikátorů, zařazených za sebou. Podstata kaskádního klasifikátoru spočívá v tom, subobraz musí projít celou kaskádou, aby byl přijat jako obličej. Pokud je vyřazen například po prvním stupni kaskády, začne prohledávání dalšího subobrazu. Tím, že jsou vypočteny jen slabé klasifikátory z první kaskády, kterých není mnoho, dojde k úspoře času.

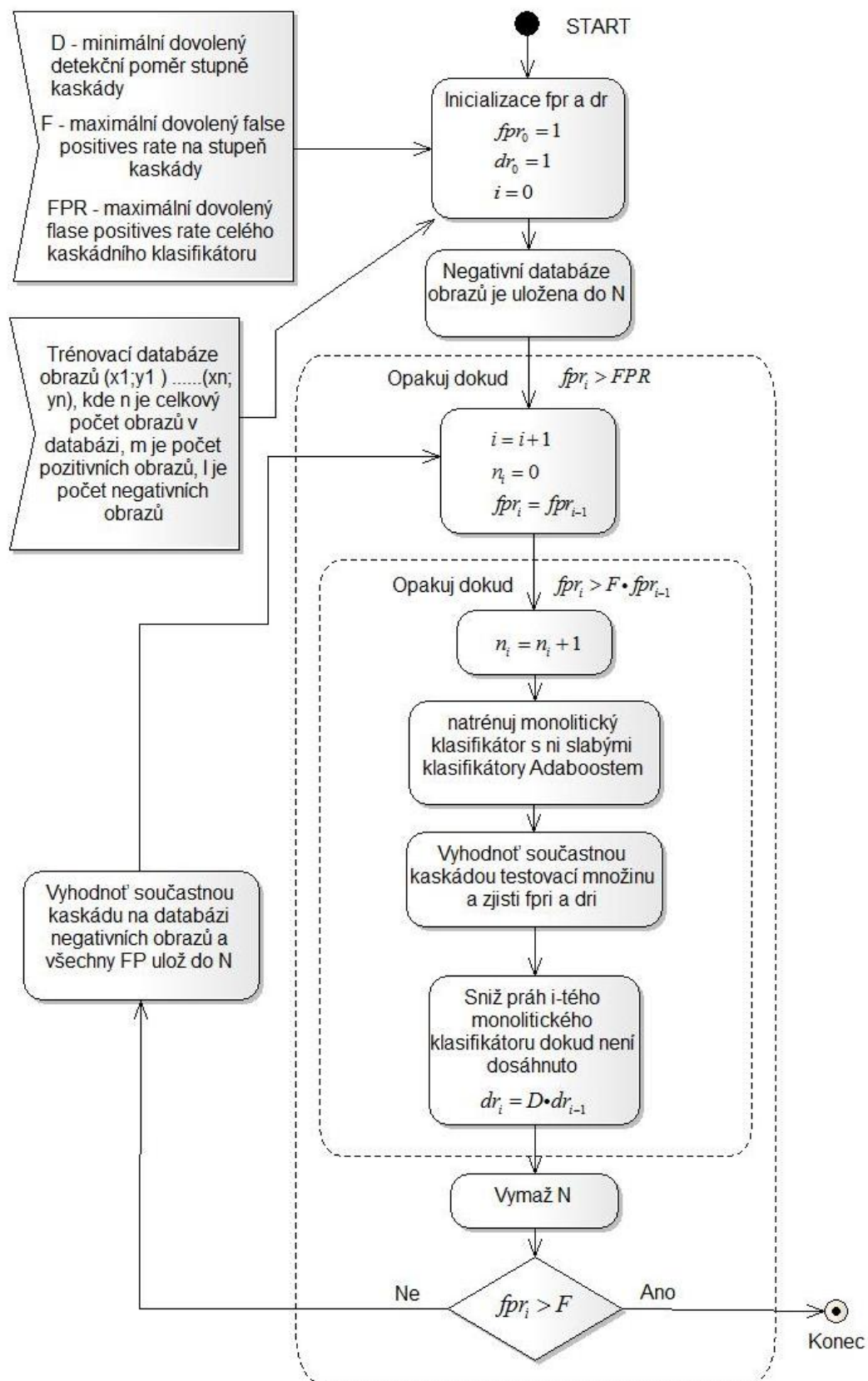
Trénování kaskádního klasifikátoru ovšem probíhá odlišně než trénování monolitického klasifikátoru. Vývojový diagram trénování kaskádního klasifikátoru je na obrázku 4.6. Nechť je vstupem algoritmu pro trénování kaskády, minimální detekční poměr pro každý stupeň kaskády  $D$  a celkový false positives rate  $FPR$

$$FPR = \frac{\text{počet FP}}{\text{počet prohledaných subobrazů}}, \quad (4.5)$$

případně počet stupňů kaskády  $K$ . Vzhledem k faktu, že kaskáda jako celek je založena na principu odmítání subobrazů, je možné stanovit maximální  $FPR$  pro celou kaskádu, přičemž platí:

$$FPR = \prod_{i=1}^K fpr_i, \quad (4.6)$$

kde  $fpr_i$  jsou false positives rate pro každý stupeň kaskády a  $K$  je celkový počet stupňů kaskády. Chceme-li tedy dosáhnout nízkého počtu false positives, pak stačí, aby každý stupeň kaskády měl  $FPR \approx 0,3$  a počet stupňů v kaskádě byl  $K \approx 10$ , pak celkový  $FPR = 0,3^{10} = 0,000006$ , což je přijatelná hodnota. Proto je použito i  $fpr_i$  pro hledání optimálního monolitického klasifikátoru pro každý stupeň kaskády. Na druhou stranu chceme dosáhnout i vysokého celkového detekčního poměru  $DR$ , takže je nutné nalézt kompromis mezi  $FPR$  a  $DR$ , toho je dosaženo tak, že při trénování kaskády je pro každý  $i$ -tý stupeň definován  $fpr_i$  a  $dr_i$  a je opakovaně trénován takový monolitický klasifikátor dokud nejsou  $fpr_i$  a  $dr_i$  splněny.



Obr. 4.6 Vývojový algoritmus trénování kaskádního klasifikátoru

## 4.6 Implementace detektoru

Pro implementaci byl zvolen postup, kdy jsou jednotlivé kroky programu řešeny sekvenčně, jsou použity výše uvedené matematické vztahy. Tento postup byl zvolen proto, že v jednotlivých krocích všech cyklů, je možné kontrolovat výpočty, vyniká více fyzikální podstata algoritmu a kód je lehce čitelný. Vývojový algoritmus je na obrázku 4.10 a jeho popis je obsahem následující kapitoly.

Pro vstupní obrazy bylo zvoleno rozlišení 800x600pixelů, protože toto rozlišení patří dnes mezi standardní a umožňuje další detailní zpracování obrazu.

Pro tuto implementaci byla využita natrénovaná data Ing. Přinosila použitá [3]. Na základě konzultace s Ing. Přinosilem bylo uvaženo, že vytvoření trénovacího algoritmu by bylo mimo rozsah této práce.

Hlavní vlákno řídí celý algoritmus. V hlavním vláknu načten vstupní obraz, jsou volány jednotlivé funkce a dochází k cyklické změně měřítka vstupního obrazu a výběru subobrazu.

## 4.7 Funkce a funkční bloky algoritmu

### 4.7.1 Hlavní vlákno

Jak již bylo zmíněno, z hlavního vlákna jsou volány všechny další používané funkce, v cyklech jsou měněny zásadní proměnné určující míru podvzorkování vstupního obrazu a polohu subobrazu v podvzorkovaném obraze.

Prvním krokem je načtení souboru, který obsahuje informace z trénovací fáze, tj. polohy Haarových filtrů, prahové hodnoty slabých klasifikátorů a prahové hodnoty silných klasifikátorů. Načte se vstupní obraz a je převeden na šedotónový.

Následuje cyklus, ve kterém se mění meze podvzorkování vstupního obrazu. Tyto meze jsou nastaveny podle rozměru obličeje v obraze, který chceme detekovat, tyto meze mají zásadní vliv na výpočetní čas. Vstupní obraz je podvzorkován.

V dalším dvojitém cyklu je vybírán subobraz z podvzorkovaného obrazu. Jedná se o výřez 19x19pixelů, který je dále zpracováván. Následuje podmínka, která vyloučí příliš světlé a příliš tmavé subobrazy podle průměrného jasu v subobraze. Dalším krokem je výpočet integrálního obrazu.

### 4.7.2 Integrální obraz

Integrální obraz je vypočten dle výše uvedených vztahů v rovnicích (4.1) a (4.2). Dále je vypočten koeficient pro normalizaci rozptylu jasu v subobraze. Tento výpočet je přejatý z [3]. Použití stejného algoritmu pro normalizaci rozptylu v jasu je nevyhnutelné z důvodu, že jsou ovlivněny hodnoty příznaků. Pokud by byl použit jiný algoritmus, byla by ztracena vazba na prahové hodnoty slabých klasifikátorů.

Koeficient pro normalizaci je vypočten z integrálního obrazu vypočteného z kvadrátu vstupního obrazu.

Koeficient je vypočten:

$$D = \sqrt{\frac{\text{integ2}(19,19) - \frac{\text{integ}^2(19,19)}{19 \cdot 19}}{19 \cdot 19}}, \quad (4.7)$$

kde  $\text{integ2}(19,19)$  je hodnota v integrálním obraze vypočteném z kvadrátu vstupního subobrazu na pozici (v matici integrálního obrazu)  $[19,19]$ ,  $\text{integ}^2(19,19)$  je kvadrát integrálního obrazu vypočteného ze vstupního subobrazu na pozici (v matici integrálního obrazu)  $[19,19]$ , součin  $19 \cdot 19$  reprezentuje váhování počtem pixelů a  $D$  je koeficient pro normalizaci rozptylu jasu.

Hodnota normalizovaného příznaku je vypočtena:

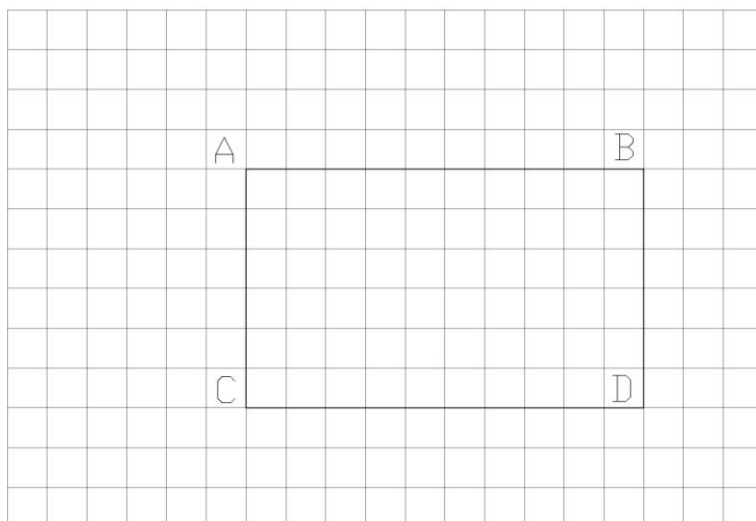
$$f(x)' = f(x) \cdot D, \quad (4.8)$$

kde  $f(x)'$  je hodnota normalizovaného příznaku,  $f(x)$  je hodnota vypočteného příznaku a  $D$  je koeficient pro normalizaci rozptylu jasu v subobraze.

### 4.7.3 Výpočet příznaků a slabé klasifikátory

Dalším krokem je výpočet odezev na Haarovy filtry - příznaků a výsledků slabých klasifikátorů.

Jsou adresovány jednotlivé údaje ze souboru, ve kterém jsou uložena natrénovaná data. To jsou především souřadnice a typy filtrů. Prvně jsou informace tříděny podle typu filtru a dále jsou do pomocných proměnných uloženy hodnoty integrálního obrazu pro výpočet odezvy.



Obr. 4.7 Výpočet příznaku – Použité pixely integrálního obrazu pro výpočet. Součet pixelů je potom,  $S = D - B - C + A$ .



Z pomocných proměnných je vypočten normalizovaný příznak podle rovnice (4.8). Tento příznak je použit pro výpočet výstupu slabých klasifikátorů, ty jsou vypočteny podle rovnice:

$$h(x, f, p, \Theta) = \text{sign}(f(x)' - \Theta), \quad (4.9)$$

kde  $\Theta$  je prahová hodnota slabých klasifikátorů stanovená na základě trénování,  $f(x)'$  je hodnota normalizovaného příznaku  $h(x, f, p, \Theta)$  je binární výsledek klasifikace slabým klasifikátorem.

Takto jsou vypočteny všechny příznaky pro daný stupeň kaskády

#### 4.7.4 Silný klasifikátor

Výsledky slabých klasifikátorů jsou předány do funkce, která realizuje silný klasifikátor. Je vypočten váhovaný součet binárních výstupů slabých klasifikátorů. Váhami  $\alpha_t$  jsou opět hodnoty stanovené trénováním. Tyto váhy jsou přímo úměrné míře informativnosti (chybovosti) příznaků. Takovýto váhovaný součet je vstupem silného klasifikátoru.

Silný klasifikátor rozhodne, zda-li se bude pokračovat ve výpočtu dalších stupňů kaskády pro daný subobraz, v případě, že nejde o poslední stupeň kaskády. V případě, že se jedná o silný klasifikátor posledního stupně kaskády, je rozhodnuto o přítomnosti obličeje. Matematicky:

$$H = \text{sign}\left(\sum_{t=1}^T \alpha_t \cdot h(x, f, p, \Theta) - P\right), \quad (4.10)$$

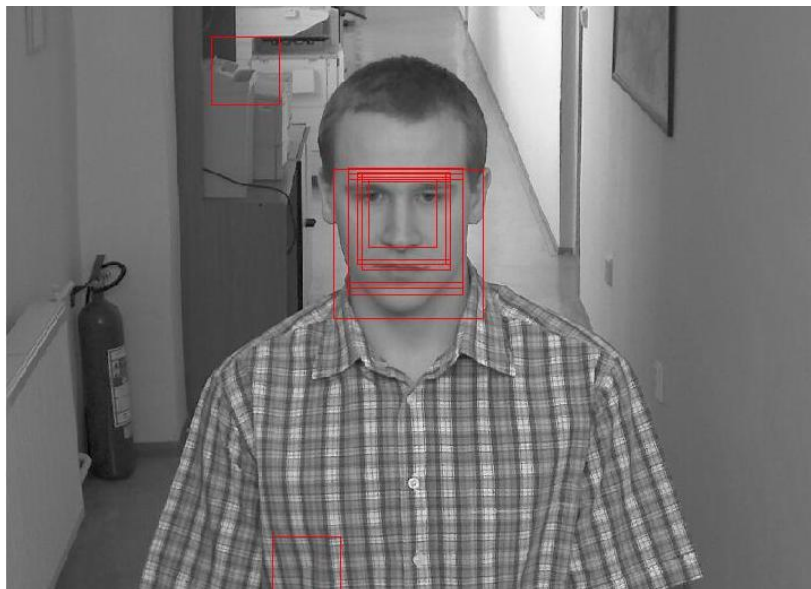
kde  $\alpha_t$  jsou váhy konkrétního slabého klasifikátoru  $h(x, f, p, \Theta)$ ,  $P$  je prahová hodnota silného klasifikátoru,  $T$  je celkový počet příznaků v kaskádě a  $H$  je binární výstup silného klasifikátoru.

V případě, že stavy všech silných klasifikátorů jsou 1, pak je subobraz přijat jako obličej a jsou uloženy souřadnice středu subobrazu v měřítku originálního vstupního obrazu a rozměr subobrazu v originálním vstupním obraze.

Tímto způsobem je prohledán celý obraz ve všech požadovaných stupních podvzorkování. Výstupem této etapy může být mnohonásobná detekce jednoho obličeje v různých měřítkách a na různých (blízkých) pozicích subobrazu.

#### 4.7.5 Shluková analýza

Nachází-li se v obraze obličej a rozměr detektoru odpovídá rozměru obličeje tak, aby detektor byl schopen obličej nalézt, pak dojde k obvyklému jevu mnohanásobné detekce jednoho obličeje. Tato skutečnost je ukázána na obrázku 4.8.



Obr. 4.8 Mnohonásobné detekce jednoho obličeje.

V této fázi algoritmu byl obličej sice nalezen, ale vznikly také dvě chyby typu false positives. Tento výsledek není uspokojivý a je zcela nepoužitelný pro případné navazující algoritmy např. rozpoznávání tváře. Z tohoto důvodu je nutné nějakým způsobem násobné detekce sloučit a odstranit false positives. K tomu je vhodná shluková analýza.

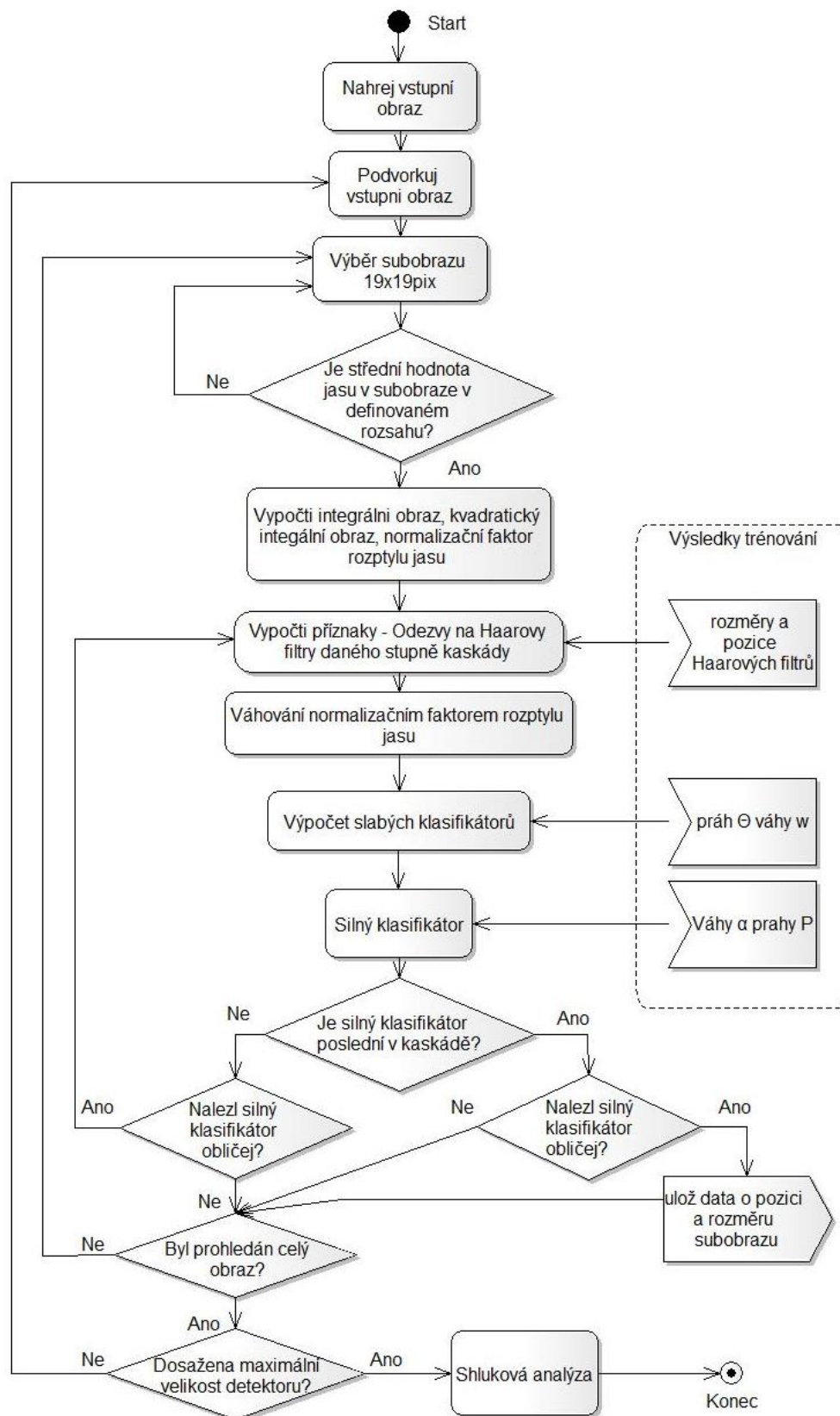
Shluková analýza je proces, během něhož, jsou jednotlivé prvky nějaké množiny seskupeny do shluků. Seskupení do shluků je vždy založeno na nějakém kritériu, podle něhož je seskupení provedeno. V tomto případě je tímto kritériem euklidovská vzdálenost v trojrozměrném prostoru. Prvním rozměrem je vertikální pozice středu detekovaného subobrazu, druhým rozměrem je horizontální pozice středu detekovaného subobrazu a třetím rozměrem je rozměr detekovaného subobrazu. Tyto vzdálenosti jsou vypočteny mezi jednotlivými detekcemi a skupina subobrazů, mezi kterými je menší vzdálenost, než zadaný práh je sloučena do shluku.

V obrázku 5.3 je tedy výsledkem odstranění dvou chyb typu false-positives a vytvoření shluku v oblasti obličeje. Takovýto výstup z klasifikačního algoritmu je již uspokojivý pro další zpracování obrazu tváře.



Obr. 4.9 Výsledek shlukové analýzy.

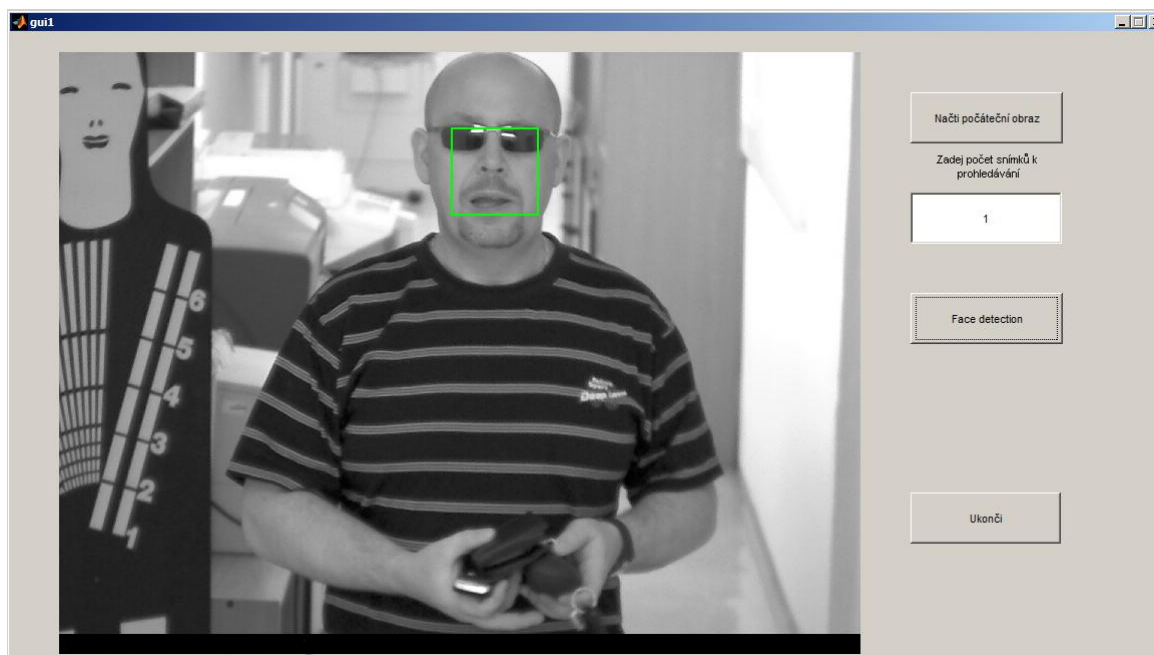
#### 4.7.6 Vývojový diagram implementace detektoru Viola - Jones



Obr. 4.10 Vývojový diagram implementace detektoru Viola-Jones

## 4.8 Uživatelské rozhraní

Pro potřeby testování detektoru a snazšího ovládání bylo vytvořeno uživatelské rozhraní, které umožňuje načíst obraz ze souboru, případně několik obrazů a provést na nich detekci obličejů.



Obr. 4.11 Uživatelské rozhraní

## 5 TESTOVÁNÍ DETEKTORU

Algoritmy pro detekci obličeje bývají testovány, aby bylo možné zjistit a porovnat jejich vlastnosti s jinými metodami. V této práci budou prezentovány výsledky dvou implementací detektoru Viola – Jones. Jeden z nich byl implementován v rámci této práce (pro potřeby této práce bude označován jako detektor TM), druhý byl implementován Vahidem Kazemim (označeno jako detektor Vahid Kazemi), obě implementace jsou naprogramovány v jazyce MATLAB. K dispozici je dále implementace detektoru Viola-Jones v jazyce C++ firmy Neurotechnology v SDK Verilook (označen jako detektor Neurotechnology). Implementace firmy Neurotechnology ovšem umožňuje zpracovat pouze data získaná z kamer. Proto nemohla být tedy podrobena všem testům v požadovaném rozsahu.

### 5.1 Vstupní předpoklady

Před testováním, byla implementace Vahida Kazemiho detailně analyzována, aby bylo možné nastavit oběma detektorům stejné parametry pro prohledávání. Z této analýzy vyplývá, že je intenzivně využíváno matic převedených na vektor vytvořený z řádků matice. Použití jednorozměrných polí je časově úspornější. Velmi výhodné je použití skalárního součinu pro výpočet příznaků. Z tohoto vyplývá, že doba zpracování jednoho obrazu bude pravděpodobně nižší než doba potřebná pro zpracování obrazu detektorem TM. Další z parametrů ovlivňující časovou náročnost obou algoritmů jsou nastavení míry podvzorkování vstupního obrazu a velikost posunu detekčního okna v obraze.

Detekční schopnosti detektorů je možné odhadnout jen velmi těžko, protože není známá trénovací množina ani trénovací algoritmus. Známa fakta jsou typy Haarových filtrů, počet příznaků a typ klasifikátoru. Implementace Vahida Kazemiho používá monolitický klasifikátor s pětistý příznaky, zatímco detektor implementovaný v rámci této práce využívá kaskádní klasifikátor s devatenácti stupni a celkovým počtem příznaků 146. Implementace Vahida Kazemiho používá stejné typy Haarových filtrů jako implementace TM. O implementaci detektoru firmy Neurotechnology není znám žádný parametr.

Pro statistické vyhodnocení definujeme následující veličiny, které shrnují výsledky testování:

True positive rate:

$$TPR = \frac{TP}{TP + FN} = \frac{TP}{\text{celkový počet obličejů}} \quad (5.1)$$

False positives rate:

$$FPR = \frac{FP}{\text{počet obrazů}} \quad (5.2)$$

False positive rate 2:

$$FPR2 = \frac{FP}{\text{počet subobrazů v celé databázi}} \quad (5.3)$$

Přesnost detekce:

$$PPV = \frac{TP}{TP + FP} \quad (5.4)$$

Některé veličiny byly již definovány v úvodu, nicméně v mnoha pracích jsou definovány různě (proto je použito FPR a FPR2), v zájmu exaktnosti byly jednoznačně definovány pro tuto statistiku.

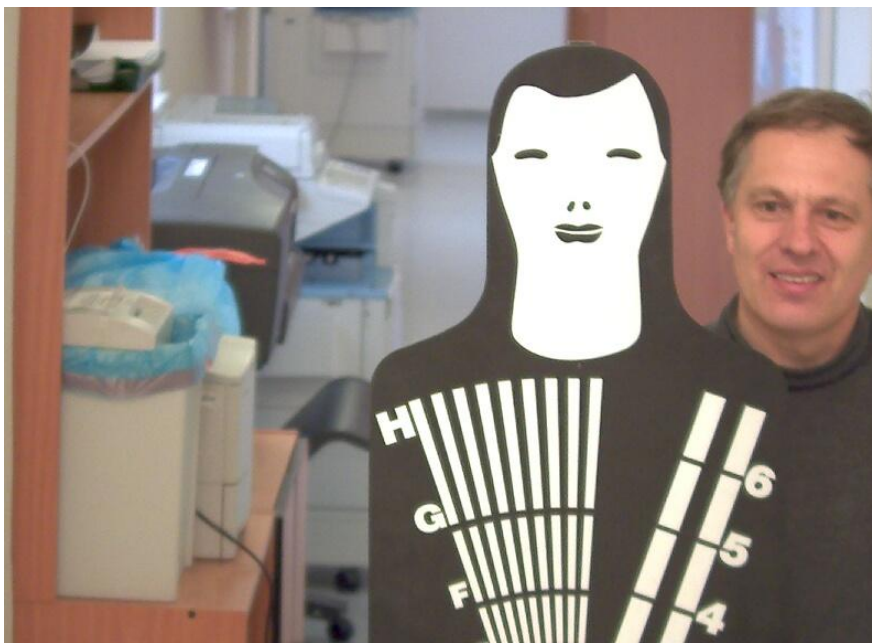
## 5.2 Testování

Algoritmy detekce obličeje je možné testovat a porovnávat na různých obrazových databázích, které obsahují různé obrazy. Tyto databáze výrazně ovlivňují výsledky testů. Proto se obvykle používají standardizované databáze pro testování, aby byly výsledky vzájemně porovnatelné. Vzhledem k obtížnější dostupnosti větších databází, byla sestrojena obecná databáze z několika menších databází. Na této databázi budou testovány detekční schopnosti algoritmu.

Dalším důležitým parametrem detektoru je odolnost detektoru proti natočení hlavy v různých osách, za tímto účelem byla sestrojena speciální databáze (označeno jako databáze k testování odolnosti proti náklonu hlavy).

### 5.3 Vlastnosti testovací databáze

Byla sestrojena databáze s 1045 obrazy. V této databázi se vyskytují i obrazy s více obličejí. Rozlišení obrazů je 800x600pix. Osvětlení scény je rovnoměrné a stabilní. V databázi se vyskytují obličejí s přirozeným výrazem a polohou. Rozměr obličejů je v rozmezí od 70 pix do 250 pix.

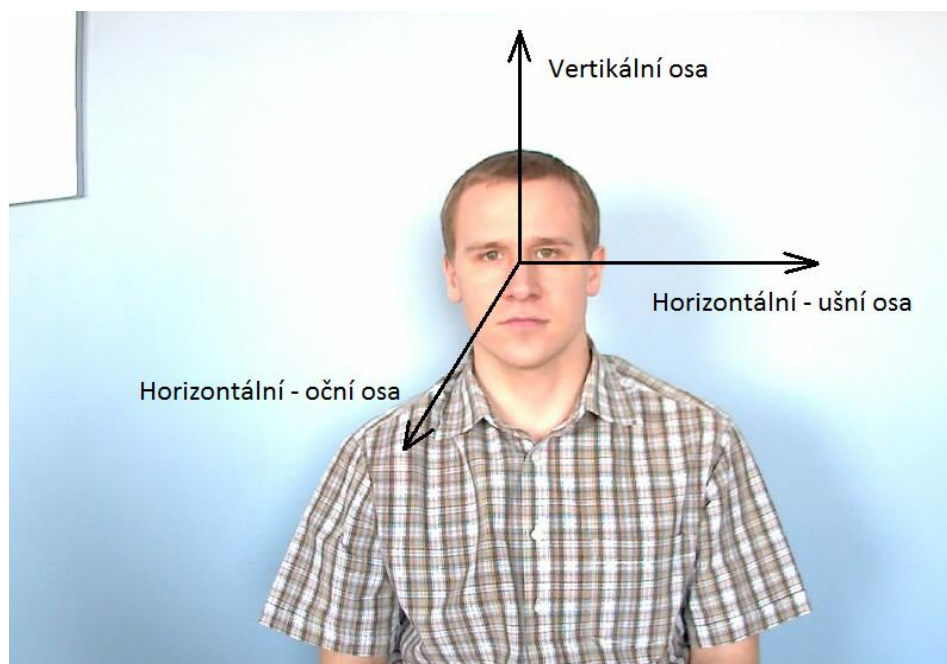


Obr. 5.1 Příklad obrázku z testovací databáze

### 5.4 Databáze pro testování odolnosti proti náklonu hlavy

Principem testování odolnosti detektoru proti náklonu hlavy je nalezení mezního úhlu natočení hlavy, kdy detektor nalezne obličej. Testování spočívá v předkládání série obrazů, kde se mění náklon hlavy po velmi malém kroku, takto je nalezen mezní úhel. Pro tento test bylo nutné sestavit speciální databázi. Pro pořízení této databáze bylo zřízeno dočasné pracoviště umístěné tak, aby scéna byla dobře osvětlená, s monolitickým pozadím. Ve scéně se vyskytuje pouze horní polovina těla člověka na jednom místě a mění se náklon hlavy. Databáze obsahuje šest různých osob, které natáčejí hlavu ve třech různých osách. Tyto osy jsou blíže specifikovány na obr. 5.2





Obr. 5.2 Specifikace os okolo kterých, se otáčí hlava na obrazech v databázi pro testování odolnosti proti naklonění hlavy.

## 5.5 Výsledky testů

### 5.5.1 Výsledky testu na databázi

Na testovací databázi byly testovány dva detektory. Detektor TM a detektor Vahida Kazemiho. Jak bylo již výše uvedeno, detektor firmy Neurotechnology neumožňuje takto nadefinovat vstup algoritmu. Celkem se v databázi vyskytuje 1075 obličejů (Některé obrazy obsahují více obličejů).

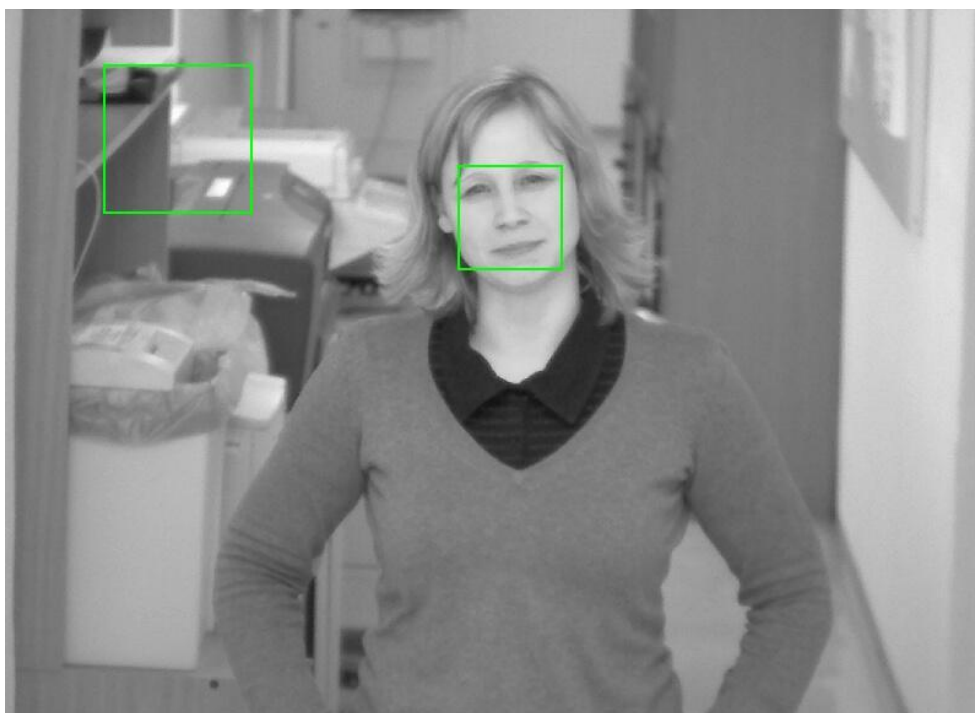
Detektor	Vahid Kazemi	TM
$\Sigma$ TP	987	916
$\Sigma$ FP	60	94
$\Sigma$ FN	87	152
<b>TPR</b>	0,91814	0,852093
<b>FPR</b>	0,05742	0,08995
<b>FPR2</b>	5,01E-07	7,84E-06
<b>PPV</b>	0,943	0,824482

Tabulka 3 Výsledky testování

### 5.5.2 Diskuze výsledků

Výsledky detektoru TM, byly značně zkresleny, neboť na cca 40-ti obrazech se projevila stejná chyba detektoru typu false positive. Tato chyba je zobrazena na obr. 6.3. Pomineme-li tuto soustavnou chybu, je počet false positives srovnatelný s počtem false positives implementace Vahida Kazemiho. V několika případech se chyba typu false positive objevila na oblečení (kostkované a kosočtvercové vzory), bylo také zaznamenáno větší množství false positives na čele osob s menším množstvím vlasů. False negatives byly mnohdy způsobeny výraznějším náklonem hlavy. False negatives u implementace TM je vyšší než u implementace Vahida Kazemiho právě pro menší odolnost proti náklonu hlavy, tato skutečnost je popsána v následující podkapitole.

Detektor implementovaný Vahidem Kazemim měl zhruba o 6 % více úspěšnějších výsledků detekce obličeje v obraze (true positives) s porovnáním detektoru TM. Jak ale bylo zmíněno v předchozích kapitolách, nejsou známá vstupní data trénování algoritmů. Víme ale, že detektor Vahida Kazemiho používá podstatně více příznaků v detekčním subobrazu, než je celkový počet příznaků v kaskádě detektoru TM. Negativem, které může do jisté míry ovlivnit parametr hodnocení - false positives je právě velký počet těchto možných příznaků v detekčním okně. Tohle negativum se projevilo zřetelně na oblečení procházejících osob s proužkovaným a kostkovaným vzorem košil s porovnáním těchto výsledků detektoru TM.



Obr. 5.3 Příklad výsledku implementace detektoru TM

### 5.5.3 Výsledky testu odolnosti detektorů proti naklonění hlavy

Tomuto testu byly podrobeny: detektor Vahida Kazemiho, detektor TM i částečně detektor Neurotechnology. Detektor firmy Neurotechnology byl testován přes webovou kameru. Číselné výsledky mezních úhlů byly u všech detektorů změřeny softwarově a to pouze při náklonu hlavy okolo horizontální oční osy. Otočení ve zbývajících dvou osách by bylo možno pouze odhadovat, proto bude nutné se spokojit s obrazovým vyjádřením mezních úhlů.

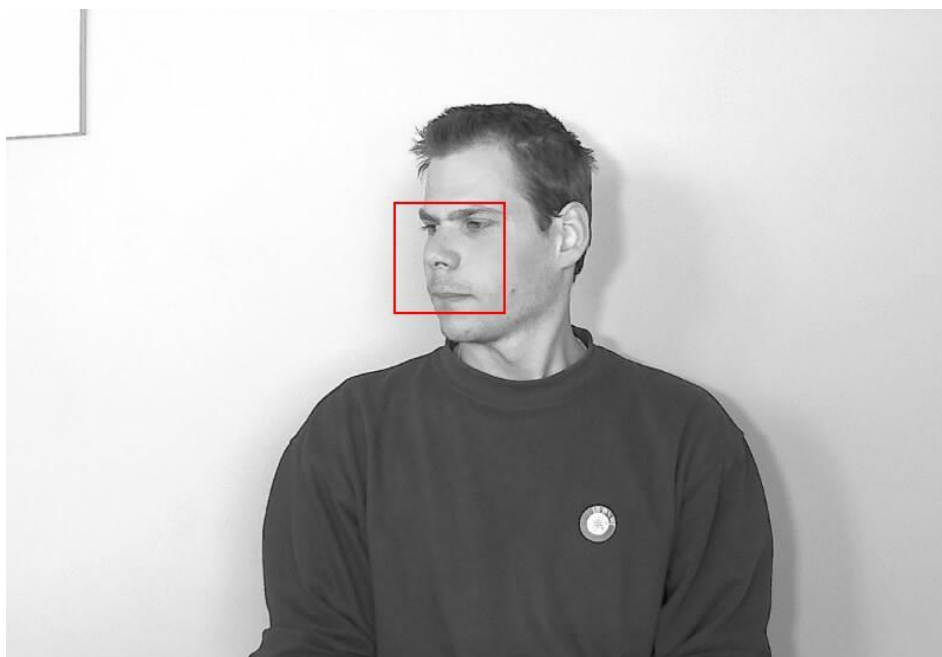
	Detektor	mezní úhly (°)						průměrný úhel (°)
levý náklon	TM	8,5	5,6	6,5	13,6			8,6
	Vahid Kazemi	17,7	12,5	19,8	5,5	16	16,8	14,7
pravý náklon	TM	7,2	8,6	3,6	16,2			8,9
	Vahid Kazemi	13,4	6	17,9	8,4	17	13,2	12,7

Tabulka 4 Mezní úhly naklonění hlavy okolo horizontální oční osy.

V tabulce 4 chybějí některá data pro detektor TM, to z toho důvodu, že některé osoby s nakloněnou hlavou nebyly vůbec detekovány. Tento problém nelze přisoudit pouze neschopnosti detektoru obličeje detekovat, ale je nutné zmínit, že osoby vyskytující se v databázi nebyly schopny zachovat standardní výraz obličeje a nebyly schopné náklonu hlavy v jedné ose. Navzdory tomu je ale možné stanovit mezní úhly naklonění hlavy alespoň přibližně. Z tabulky 4 vyplývá, že detektor Vahida Kazemiho je odolnější proti náklonu hlavy v horizontální oční ose. Detektor Neurotechnology je schopen detekovat obličeje i při úhlu natočení 90°.

Mezní úhly v dalších dvou osách je možné pouze porovnat, nikoliv změřit. Detektor TM prakticky není odolný proti otočení hlavy kolem vertikální osy. Navzdory tomu detektor je detektor Vahida Kazemiho schopen detekovat obličej i ve značném natočení. Detektor Neurotechnology má při natočení hlavy kolem vertikální osy podobně detekční schopnosti jako detektor Vahida Kazemiho.

Při porovnání mezních úhlů natočení hlavy okolo horizontální ušní osy jsou výsledky všech detektorů srovnatelné. Při záklonu hlavy je detektor TM méně odolnější než další dva detektory.



Obr. 5.4 Výsledek testu natočení hlavy okolo vertikální osy detektoru Vahida Kazemiho

## 5.6 Výpočetní čas a efektivita

Výpočetní čas je určen programovacím jazykem a implementací. Vzhledem k tomu, že SDK firmy Neurotechnology patří mezi nejsilnější nástroje pro detekci obličeje a rozpoznávání obličeje na světě, je jeho čas řádově nižší než výpočetní dalších dvou implementací. Tento čas nebylo možné přesně změřit.

U dalších dvou implementací, byly podle potřeby měněny mezní rozměry detektoru a velikost změn rozměru. Rozsah, ve kterém je rozměr detektoru měněn má zcela zásadní vliv na výpočetní čas. Implementace Vahida Kazemiho využívá operace s vektory, což celou implementaci činí velmi efektivní. Při rozsahu rozměru detektoru cca 70 – 250pix., s celkový počtem čtrnácti rozměrů (použito při testování na databázi průchodu) byl výpočetní čas detektoru TM cca 17sekund, zatímco detektor Vahida Kazemiho dosahoval výpočetního času 7,5 sekundy.

## 6 ZÁVĚR

V úvodu této práce byly stanoveny dva hlavní cíle této práce. Jedním z nich je seznámit se s problematikou detekce tváře a metodami pro detekci obličeje v obraze, druhým z nich je implementovat detektor Viola-Jones v jazyce Matlab.

V druhé kapitole byl popsán princip detekce tváře a vysvětleny základní termíny z dané problematiky. Byla představena obecná kriteria, podle kterých je možné hodnotit metody pro detekci obličeje.

Třetí kapitola je věnována popisu metod detekce obličeje v obraze. Jednotlivé metody byly rozděleny do skupin podle principu, na kterém jsou založeny. Zvláštní pozornost byla věnována metodám založených na vzhledu, protože tyto metody využívají obvykle nelineární klasifikátory, což vede k lepším výsledkům. Charakteristickou vlastností těchto metod je i nižší výpočetní doba oproti většině ostatních metod. Na základě tohoto přehledu byla vybrána metoda, která bude realizována. Tato metoda patří právě do skupiny metod založených na vzhledu. Jedná se o detektor Viola Jones. Metoda je teoreticky popsána v kapitole čtyři. V závěru třetí kapitoly jsou shrnuta dostupná objektivní data pro vyhodnocení a porovnání charakteristik jednotlivých metod.

Čtvrtá kapitola pojednává o vybraném detektoru Viola-Jones. Důvody pro jeho výběr jsou zejména velmi dobré detekční schopnosti při nízké výpočetní době. Dalším důvodem pro výběr této metody je fakt, že je k dispozici velice omezené množství konkrétních informací o konstrukci a trénování tohoto algoritmu. Ve čtvrté kapitole je detektor Viola-Jones detailně analyzován a jsou popsány jednotlivé dílčí kroky algoritmu. Popsán je i trénovací algoritmus Adaboost, pomocí něhož jsou vybrány informativní příznaky.

Po analýze celého procesu implementace detektoru Viola-Jones, se autor po konzultaci s vedoucím práce dohodl, že zcela kompletní implementace detektoru byla mimo rozsah této práce. Z toho důvodu nebyl realizován trénovací algoritmus Adaboost a natrénována data byla převzata od Ing. Přinosila. Za použití těchto dat byl kompletně implementován detektor Viola-Jones s kaskádním klasifikátorem. První funkční verze algoritmu byla ovšem značně výpočetně náročná, proto byla několikrát částečně přepracována a upravena, zahrnuty byly i některé podmínky, které algoritmus urychlily. Nejvíce časově náročná část algoritmu je výpočet příznaků, ta tvoří cca 70% výpočetního času. Výpočetní čas byl během optimalizací snížen na 17 sekund na obraz o rozměru 800x600pixelů, což představuje 24% původního času.

Pátá kapitola obsahuje přehled výsledků různých implementací detektoru Viola-Jones. Z výsledků testu na obecné databázi můžeme usoudit, že detektor implementovaný v rámci této práce má dobré detekční schopnosti a může konkurovat jiným neprofesionálním implementacím. Detekční poměr TPR je 85% za současně nízkého počtu falešných detekcí FPR 8%. Na základě výsledků páté kapitoly je možné konstatovat, že byl vytvořen plně funkční algoritmus pro detekci obličeje v obraze.

## 7 LITERATURA

- [1] VIOLA, Paul; JONES, Michael. *Robust Real-time object detection*. Vancouver, Canada : [s.n.], 2001. 25 s.
- [2] VIOLA, Paul; JONES, Michael. Robust Real-Time Face Detection. In *International Journal of Computer Vision* . The Netherlands : Kluwer Academic Publishers, 2003. s. 137-154.
- [3] PŘINOSIL, Jiří. *Analýza emocionálních stavů na základě obrazových předloh*. Brno, 2008. 90 s. Dizertační práce. Vysoké učení technické v Brně.
- [4] JÁNOŠ, Jiří. *Face detection in images*. Brno, 2008. 79 s. Bakalářská práce. Masarykova Univerzita.
- [5] LIENHART, Rainer; MAYDT, Jochen. An Extended Set of Haar-like Features for Rapid Object Detection. *IEEE ICIP*. 2002, s. 4.
- [6] MITA, Takeshi ; KANEKO, Toshimitsu; HORI, Osamu. Joint Haar-like Features for Face Detection. *IEEE International Conference on Computer vision*. 2005, s. 8.
- [7] CONSTANTINE, Papageorgiou; OREN, Michael; POGGIO, Tomaso. A General Framework for Object Detection. *IEEE International Conference on Computer vision*. 1998, s. 555-562.
- [8] JENSEN, Ole Helvig. *Implementing the Viola-Jones Face Detection Algorithm*. Kongens Lyngby, Denmark : Technical University of Denmark, 2008. 35 s.
- [9] CHEN, Y. J., LIN, Y., C. Simple Face-detection Algorithm Based on Minimum Facial Features. The 33rd Annual Conference of the IEEE Industrial Electronic Society IECON 2007. Taipei, 2007, p. 455-460.
- [10] MARIUS, Diedrick ; PENNATHUR, Sumita ; ROSE, Klint . Face Detection Using Color Thresholding, and Eigenimage Template Matching. [cit. 2010-10-15]. Dostupný z WWW: <[http://www.stanford.edu/class/ee368/Project\\_03/Pro](http://www.stanford.edu/class/ee368/Project_03/Pro)>.
- [11] PONZER, Martin . *DETEKCE A ROZPOZNÁVÁNÍ OBLICEJE* [online]. Brno 2009. 84 s. Diplomová práce. VUT v Brně. Dostupné z WWW: <[https://www.vutbr.cz/www\\_base/zav\\_prace\\_soubor\\_verejne.php?file\\_id=14688](https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=14688)>.
- [12] TURK, M.; PENTLAND, A. Eigenfaces for Face Detection/Recognition. In *Journal of Cognitive Neuroscience* [online]. 1991 [cit. 2010-10-20]. Dostupné z WWW: <[http://www.vision.jhu.edu/teaching/vision08/Handouts/case\\_study\\_pca1.pdf](http://www.vision.jhu.edu/teaching/vision08/Handouts/case_study_pca1.pdf)>.
- [13] KROLIKOWSKI, Martin. *AUTOMATICKÉ DETEKCE OBLICEJE A JEHO JEDNOTLIVÝCH ČÁSTÍ* [online]. Brno 2008. 67 s. Diplomová práce. VUT v Brně. Dostupné z WWW: <[https://www.vutbr.cz/www\\_base/zav\\_prace\\_soubor\\_verejne.php?file\\_id=8173](https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=8173)>.
- [14] YANG, Ming-Hsuan; KRIEGMAN, David J. ; AHUJA, Narendra. Detecting Faces in Images: A Survey. In *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*. 2002. s. 25.
- [15] KOTROPOULOS, Constantine ; PITAS, Ioannis . Rule-based face detection in frontal views. [online]. [cit. 2010-10-30]. Dostupný z WWW: <[http://www.ece.umassd.edu/Faculty/acosta/ICASSP/Icassp\\_1997/pdf/author/ic972537.pdf](http://www.ece.umassd.edu/Faculty/acosta/ICASSP/Icassp_1997/pdf/author/ic972537.pdf)>.

- [16] YANG, G. ; HUANG, T. S..Human Face Detection in Complex Background. In *Pattern Recognition*, vol. 27, no. 1, pp. 53-63, 1994.
- [17] YOW, Kin Choong ; CIPOLLA, Roberto A Probabilistic Framework for Perceptual Grouping of Features for Human Face Detection. In *A Probabilistic Framework for Perceptual Grouping of Features for Human Face Detection* [online]. Cambridge, UK : University of Cambridge, 79 [cit. 2010-11-03]. Dostupné z WWW: <<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.28.7082>>.
- [18] PING, Scott Tan Yeh ; WENG, Chun Hui ; LAU, Boonping . *FACE DETECTION THROUGH TEMPLATE MATCHING AND COLOR SEGMENTATION* [online]. Nevim : Nevim, 89 [cit. 2010-10-16]. Dostupné z WWW: <[http://www.stanford.edu/class/ee368/Project\\_03/Project/reports/ee368group04.pdf](http://www.stanford.edu/class/ee368/Project_03/Project/reports/ee368group04.pdf)>.
- [19] HAN, Chin-Chuan, et al. Fast face detection via morphology-based pre-processing.[online]. [cit. 2010-11-10]. Dostupný z WWW: <<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.29.4448>>.
- [20] SUNG, Kah-Kay ; POGGIO, Tomaso . Example-Based Learning for View-Based Human Face Detection. [online]. 1998, [cit. 2010-11-12]. Dostupný z WWW: <<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.17.1283>>.
- [21] MOGHADDAM , Baback; PENTLAND, Alex . Probabilistic visual learning for object recognition. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE* [online]. 1997, 19, [cit. 2010-11-13]. Dostupný z WWW: <<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.38.5998>>.
- [22] ROWLEY, Henry A. ; BALUJA, Shumeet; KANADE, Takeo . Neural Network-Based Face Detection. [online]. 1998, [cit. 2010-11-13]. Dostupný z WWW: <<http://www.cmucam.org/raw-attachment/wiki/viola-jones/rowley-ieee.pdf>>.
- [23] BRIMBLECOMBE, Phil. *Face Detection using Neural Networks* [online]. University of Surrey, 83 s. Projekt. University of Surrey. Dostupné z WWW: <[http://personal.ee.surrey.ac.uk/Personal/T.Windeatt/ug\\_projects/brimblecombe/phil.pdf](http://personal.ee.surrey.ac.uk/Personal/T.Windeatt/ug_projects/brimblecombe/phil.pdf)>.
- [24] OSUNA, Edgar ; FREUD, Robert ; GIROSI, Federico. Training support vector machines: An application for face detection. [online]. 1997,[cit. 2010-11-14]. Dostupný z WWW: <<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.9.6021>>.
- [25] Masarykova univerzita - : *Studijní materiály předmětu FI:PA034* [online]. 2006, 14.11.2010 [cit. 2010-11-14]. Support vector machines. Dostupné z WWW: <[http://is.muni.cz/el/1433/podzim2006/PA034/09\\_SVM.pdf?fakulta=1433;obdobi=3523;kod=PA034](http://is.muni.cz/el/1433/podzim2006/PA034/09_SVM.pdf?fakulta=1433;obdobi=3523;kod=PA034)>.
- [26] LIBOSVÁR, Jakub. *Neuronové sítě pro lokalizaci lidského obličeje* [online]. Brno , 2009. 44 s. Bakalářská práce. VUT v Brně. Dostupné z WWW: <<http://www.fit.vutbr.cz/study/DP/rpfile.php?id=6298>>.
- [27] VLACH, Jan ; PŘINOSIL, Jiří . Lokalizace obličeje v obraze s komplexním pozadím. *Elektrorevue* [online]. 11.4.2007, 2007/12, [cit. 2010-11-19]. Dostupný z WWW: <<http://www.elektrorevue.cz/cz/download/presne-sledovani-pohybu-tvari-v-realnem-case/>>.
- [28] VANSÁ, Tibor . *Implementace algoritmu pro transformaci acyklického orientovaného grafu na esenciální graf* [online]. Praha : Univerzita Karlova v Praze, 2006. 24 s. Bakalářská práce. Univerzita Karlova v Praze. Dostupné z WWW: <<http://atrey.karlin.mff.cuni.cz/~simecek/skola/Vansa.pdf>>.

- [29] PŘINOSIL, Jiří . Přesné sledování pohybu tváře v reálném čase. *Elektrorevue*. 20.11.2009, 2009/33, s. 5. ISSN 1213-1539.
- [30] THEODORIDIS, Sergios; KOUTROUMBAS, Konstantinos. *Pattern Recognition*. 4th edition. [s.l.] : Academic Press, 2009. 961 s. ISBN 978-1-597-59749-272-0
- [31] THEODORIDIS, Sergios , et al. *Introduction to Pattern Recognition : A Matlab approach*. [s.l.] : Academic Press, 2010. 219 s. ISBN 978-0-12-374486-9.
- [32] GONZALES, Rafael C.; WOODS, Richard E. *Digital Image Processing*. 3rd edition. [s.l.] : Pearson Prentice Hall, 2008. 954 s. ISBN 978-0-13-168728-8.
- [33] DINI, Fabrizio . *Media Integration and Communication Center* [online]. 2008 [cit. 2010-07-29]. An application of Viola-Jones algorithm: face detection and tracking. Dostupné z WWW: <<http://www.micc.unifi.it/dini/download/dbmm2008-Dini.pdf>>.



## 8 SEZNAM ZKRATEK A DEFINIC

<b>Fps</b>	(frames per second) počet obrazů zpracovaných za sekundu.
<b>WWW</b>	(World wide web) internetová síť
<b>FP</b>	false positives
<b>DR</b>	detection rate – detekční poměr
<b>FPR</b>	false positives rate
<b>FPR</b>	false positives rate
<b>TP</b>	True positives – správná detekce obličeje
<b>FN</b>	False negatives – nenalezení tváře, která se v obraze vyskytuje
<b>TPR</b>	$TPR = \frac{TP}{TP + FN} = \frac{TP}{\text{celkový počet obličejů}}$
<b>FPR</b>	$FPR = \frac{FP}{\text{počet obrazů}}$
<b>FPR2</b>	$FPR2 = \frac{FP}{\text{počet subobrazů v celé databázi}}$
<b>PPV</b>	$PPV = \frac{TP}{TP + FP}$

## 9 GLOSÁŘ

<b>HSV</b>	(Hue Saturation Value) je barevný model sestávající ze tří složek. H(hue) tón barvy, S(Saturation) sytost barvy, V(value) a jas [28].
<b>YCbCr</b>	Je barevný model sestávající ze tří složek. Y - jas, Cb - modrý chrominční component, Cr – červený chrominční component [29].
<b>FERET</b>	internetová databáze obrazů obličejů. Více na: <a href="http://www.frvt.org/FERET/default.htm">http://www.frvt.org/FERET/default.htm</a>
<b>Bayesova síť</b>	Orientovaný acyklický graf. Obsahuje vrcholy a hrany mezi vrcholy. Každý vrchol představuje náhodnou veličinu. Hrany vyjadřují závislostní vztahy mezi vrcholy[32]. Slouží k rychlému výpočtu celkové podmíněné pravděpodobnosti.

<b>False positives rate</b>	Je poměr mezi false positives a počtem obrazů neobsahujících obličeje.
<b>Cluster</b>	anglicky shluk, seskupení
<b>Clusterování</b>	Sloučení stejných objektů do útvaru (clusteru).
<b>Příznak</b>	Příznakem je cokoliv, co charakterizuje objekt a jeho vlastnosti. Příkladem příznaku může být barva objektu nebo vzdálenost jeho reprezentace v mnohorozměrném prostoru od referenčního bodu.
<b>Klasifikátor</b>	Klasifikátor je algoritmus, který je schopen rozčlenit vstupní data do předem definovaných skupin.
<b>False positives</b>	False positives je taková chyba, kdy jsou jako obličeje označeny oblasti obrazu, kde se tvář nevyskytuje, označuje se jako halucinace.
<b>Detekční poměr - TPR</b>	Detekční poměr je poměr počtu správně detekovaných obličejů algoritmem a počtu obličejů určených člověkem.
<b>False negatives</b>	False negatives je chyba, při níž jsou obličeje nacházející se v obraze klasifikovány špatně – jako pozadí.

# SEZNAM PŘÍLOH

<b>A</b>	<b>Obrazové přílohy</b>	<b>66</b>
A.1	Úspěšné detekce.....	66
A.2	Chybné detekce.....	67

## A OBRAZOVÉ PŘÍLOHY

### A.1 Úspěšné detekce



## A.2 Chybné detekce

